



## (12)发明专利

(10)授权公告号 CN 103999049 B

(45)授权公告日 2017. 10. 13

(21)申请号 201280058166.8

(22)申请日 2012.11.28

(65)同一申请的已公布的文献号

申请公布号 CN 103999049 A

(43)申请公布日 2014.08.20

(30)优先权数据

13/306,506 2011.11.29 US

(85)PCT国际申请进入国家阶段日

2014.05.27

(86)PCT国际申请的申请数据

PCT/US2012/066800 2012.11.28

(87)PCT国际申请的公布数据

W02013/082119 EN 2013.06.06

(73)专利权人 国际商业机器公司

地址 美国纽约

(72)发明人 R·N·张 崔美化 江页希

M·雷克尔 彭昶兴 H·H·谢赫

E·C·索 陶涛

(74)专利代理机构 北京市中咨律师事务所

11247

代理人 于静 张亚非

(51)Int.Cl.

G06F 9/455(2006.01)

(56)对比文件

US 2010/0199285 A1,2010.08.05,

US 2010/0083248 A1,2010.04.01,

CN 102004671 A,2011.04.06,

审查员 柏娟花

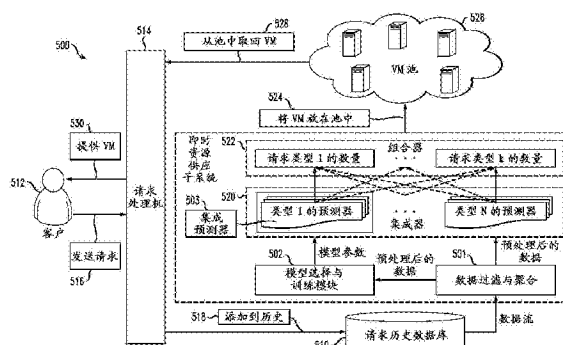
权利要求书3页 说明书19页 附图20页

### (54)发明名称

用于预测虚拟机需求的方法和装置

### (57)摘要

从云计算环境的虚拟机供应请求流中提取高级特征。针对多个不同类型的虚拟机生成多个预测模型。将回归-相关集成应用于所述预测模型中的适当模型,以便预测所述云计算环境的未来虚拟机需求。



1. 一种用于预测虚拟机需求的方法,包括:  
从云计算环境的虚拟机供应请求流中提取高级特征;  
针对多个不同类型的虚拟机生成多个预测模型;以及  
将回归-相关集成应用于所述预测模型中的适当模型,以便预测所述云计算环境的未来虚拟机需求。
2. 根据权利要求1的方法,还包括根据所预测的未来虚拟机需求,在所述云计算环境中预供应多个虚拟机。
3. 根据权利要求2的方法,其中所述提取包括过滤不相关的特性和记录以及将所述高级特征变换为时间序列。
4. 根据权利要求3的方法,其中所述过滤滤除虚拟机类型和请求时间之外的特征。
5. 根据权利要求3的方法,其中基于以不同方式处理预测不足成本和预测过度成本的非对称成本测量来执行所述生成步骤。
6. 根据权利要求5的方法,其中基于云预测成本执行所述生成步骤。
7. 根据权利要求3的方法,还包括当所述虚拟机供应请求流在时间上向前移动时,重复所述提取和生成步骤。
8. 根据权利要求3的方法,其中所述应用步骤包括:  
经由加权后的线性组合来组合多个所述预测模型以便获得中间预测;以及  
经由所述时间序列中的正相关序列之间的相关性对所述中间预测进行后处理,以便获得所预测的未来虚拟机需求。
9. 根据权利要求3的方法,还包括根据所预测的未来虚拟机需求,在所述云计算环境中预留预供应的虚拟机中的空闲虚拟机。
10. 根据权利要求1的方法,还包括提供一种系统,其中所述系统包括不同软件模块,每个所述不同软件模块包含在计算机可读存储介质中,并且其中所述不同软件模块包括原始数据过滤与聚合模块、模型生成模块以及需求预测模块;  
其中:  
由在至少一个硬件处理器上执行的所述原始数据过滤与聚合模块执行所述提取;  
由在所述至少一个硬件处理器上执行的所述模型生成模块执行所述生成;以及  
由在所述至少一个硬件处理器上执行的所述需求预测模块执行所述应用。
11. 一种用于预测虚拟机需求的计算机可读存储介质,所述计算机可读存储介质具有随其包含的计算机可读程序代码,当所述计算机可读程序代码被处理器执行时实现下列操作:  
从云计算环境的虚拟机供应请求流中提取高级特征;  
针对多个不同类型的虚拟机生成多个预测模型;以及  
将回归-相关集成应用于所述预测模型中的适当模型,以便预测所述云计算环境的未来虚拟机需求。
12. 根据权利要求11的计算机可读存储介质,其中当所述计算机可读程序代码被处理器执行时进一步实现根据所预测的未来虚拟机需求,在所述云计算环境中预供应多个虚拟机。
13. 根据权利要求12的计算机可读存储介质,其中所述提取包括被配置为过滤不相关

的特性和记录以及将所述高级特征变换为时间序列。

14. 根据权利要求13的计算机可读存储介质,其中所述过滤滤除虚拟机类型和请求时间之外的特征。

15. 一种用于预测虚拟机需求的装置,包括:

存储器;以及

至少一个处理器,其耦合到所述存储器并可操作以:

从云计算环境的虚拟机供应请求流中提取高级特征;

针对多个不同类型的虚拟机生成多个预测模型;以及

将回归-相关集成应用于所述预测模型中的适当模型,以便预测所述云计算环境的未来虚拟机需求。

16. 根据权利要求15的装置,其中所述至少一个处理器还可操作以根据所预测的未来虚拟机需求,在所述云计算环境中预供应多个虚拟机。

17. 根据权利要求16的装置,其中所述至少一个处理器可操作以通过过滤不相关的特性和记录以及将所述高级特征变换为时间序列而提取所述高级特征。

18. 根据权利要求17的装置,其中所述至少一个处理器可操作以通过滤除虚拟机类型和请求时间之外的特征来进行过滤。

19. 根据权利要求17的装置,其中所述至少一个处理器可操作以基于以不同方式处理预测不足成本和预测过度成本的非对称成本测量来生成所述多个预测模型。

20. 根据权利要求19的装置,其中所述至少一个处理器可操作以基于云预测成本生成所述多个预测模型。

21. 根据权利要求17的装置,其中所述至少一个处理器还可操作以便当所述虚拟机供应请求流在时间上向前移动时,重复所述提取和生成。

22. 根据权利要求17的装置,其中所述至少一个处理器可操作以通过以下操作应用所述回归-相关集成:

经由加权后的线性组合来组合多个所述预测模型以便获得中间预测;以及

经由所述时间序列中的正相关序列之间的相关性对所述中间预测进行后处理,以便获得所预测的未来虚拟机需求。

23. 根据权利要求17的装置,其中所述至少一个处理器还可操作以根据所预测的未来虚拟机需求,在所述云计算环境中预留预供应的虚拟机中的空闲虚拟机。

24. 根据权利要求15的装置,还包括多个不同软件模块,每个所述不同软件模块包含在计算机可读存储介质中,并且其中所述不同软件模块包括原始数据过滤与聚合模块、模型生成模块以及需求预测模块;

其中:

所述至少一个处理器可操作以通过执行所述原始数据过滤与聚合模块进行提取;

所述至少一个处理器可操作以通过执行所述模型生成模块进行生成;以及

所述至少一个处理器可操作以通过执行所述需求预测模块进行应用。

25. 一种用于预测虚拟机需求的装置,包括:

用于从云计算环境的虚拟机供应请求流中提取高级特征的部件;

用于针对多个不同类型的虚拟机生成多个预测模型的部件;以及

用于将回归-相关集成应用于所述预测模型中的适当模型,以便预测所述云计算环境的未来虚拟机需求的部件。

## 用于预测虚拟机需求的方法和装置

- [0001] 政府权利声明  
[0002] 不适用。  
[0003] 相关申请的交叉引用  
[0004] 不适用。

### 技术领域

- [0005] 本发明涉及电、电子和计算机领域,更具体地说,涉及云计算等。

### 背景技术

- [0006] 云计算力求在需要时按需提供计算资源。最新虚拟机 (VM) 供应技术可以在数十分钟内供应VM。

### 发明内容

- [0007] 本发明的原理提供用于云供应加速器的技术。在一个方面,一种示例性方法包括以下步骤:从云计算环境的虚拟机供应请求流中提取高级特征;针对多个不同类型的虚拟机生成多个预测模型;以及将回归-相关集成 (regression-correlation ensemble) 应用于所述预测模型中的适当模型,以便预测所述云计算环境的未来虚拟机需求。

- [0008] 如在此使用的,“促进”操作包括执行所述操作,使得所述操作更容易,有助于执行所述操作,或者导致执行所述操作。因此,作为实例而非限制,在一个处理器上执行的指令可以促进在远程处理器上执行的指令执行的操作,方法是发送适当的数据或命令以便导致或帮助执行所述操作。为避免疑义,在操作者通过执行操作之外的方法促进所述操作的情况下,所述操作仍由某个实体或实体组合执行。

- [0009] 本发明的一个或多个实施例或其元素可以以计算机程序产品的形式实现,所述计算机程序产品包括计算机可读存储介质,所述计算机可读存储介质具有计算机可用程序代码以便执行指示的方法步骤。此外,本发明的一个或多个实施例或其元素可以以系统(或装置)的形式实现,所述系统(或装置)包括存储器和至少一个处理器,所述至少一个处理器耦合到所述存储器并可操作以执行示例性方法步骤。更进一步,在另一个方面,本发明的一个或多个实施例或其元素可以以部件的形式实现,以便执行在此描述的一个或多个方法步骤;所述部件可以包括:(i) 硬件模块(多个),(ii) 软件模块(多个),其存储在计算机可读存储介质(或多个此类介质)中并在硬件处理器上实现,或者(iii) 所述(i)和(ii)的组合;(i)-(iii)中的任何一个都实现在此给出的特定技术。

- [0010] 本发明的技术可以提供显著的有利技术效果。例如,一个或多个实施例可以提供以下一个或多个优点:

- [0011] ●能够使云用户获得虚拟机以满足即时和紧急的能力需要,当计算工作负载增加(例如,网站上的更多点击和/或交易)时发生这些需要。

- [0012] ●能够使云提供者在非高峰时间安排供应相关工作,并且减少对备用劳动力的需

要。

[0013] 从以下将结合附图阅读的对本发明的示例性实施例的详细描述,本发明的这些和其它特性和优点将变得显而易见。

## 附图说明

- [0014] 图1示出根据本发明的一个实施例的云计算节点;
- [0015] 图2示出根据本发明的一个实施例的云计算环境;
- [0016] 图3示出根据本发明的一个实施例的抽象模型层;
- [0017] 图4示出VM需求随时间的示例性时间序列;
- [0018] 图5示出根据本发明的一个方面的示例性系统框架;
- [0019] 图6包括根据本发明的一个方面的用于预测质量的简单成本函数的符号表;
- [0020] 图7列出根据本发明的一个方面的示例性预测技术;
- [0021] 图8示出同一VM类型在数周、数天和数小时内的时间序列;
- [0022] 图9提供具有变异系数(CV)、偏度和峰度下的VM需求时间序列特征的表;
- [0023] 图10示出时间-类型-计数视图中的请求分布;
- [0024] 图11A示出对应的累积分布函数(CDF),图11B根据其请求频率对VM进行排序;
- [0025] 图12A和12B示出平均供应时间减少和平均资源浪费;
- [0026] 图13提供示出所有预测器在所有时间序列上的准确度的表;
- [0027] 图14A-14C示出三个最常见VM类型的实际时间序列和所有预测器的对应预测结果;
- [0028] 图15提供示出各种成本测量指导的集成预测器的时间减少和闲置资源的表;
- [0029] 图16提供示出可以通过结合预留控制器实现的供应时间减少的比率的表;
- [0030] 图17示出 $\beta$ 效应;
- [0031] 图18示出 $R_{fix}$ 对预测的影响(使用规范化后的数据);以及
- [0032] 图19示出根据本发明的一个方面的示例性预供应决策步骤。

## 具体实施方式

[0033] 云计算是一种服务交付模式,用于对共享的可配置计算资源池进行方便、按需的网络访问。可配置计算资源是能够以最小的管理成本或与服务提供者进行最少的交互就能快速部署和释放的资源,例如可以是网络、网络带宽、服务器、处理、内存、存储、应用、虚拟机和服务。这种云模式可以包括至少五个特征、至少三个服务模型和至少四个部署模型。

[0034] 特征包括:

[0035] 按需自助式服务:云的消费者在无需与服务提供者进行人为交互的情况下能够单方面自动地按需部署诸如服务器时间和网络存储等的计算能力。

[0036] 广泛的网络接入:计算能力可以通过标准机制在网络上获取,这种标准机制促进了通过不同种类的瘦客户机平台或厚客户机平台(例如移动电话、膝上型电脑、个人数字助理PDA)对云的使用。

[0037] 资源池:提供者的计算资源被归入资源池并通过多租户(multi-tenant)模式服务于多重消费者,其中按需将不同的实体资源和虚拟资源动态地分配和再分配。一般情况下,

消费者不能控制或甚至并不知晓所提供的资源的确切位置,但可以在较高抽象程度上指定位置(例如国家、州或数据中心),因此具有位置无关性。

[0038] 迅速弹性:能够迅速、有弹性地(有时是自动地)部署计算能力,以实现快速扩展,并且能迅速释放来快速缩小。在消费者看来,用于部署的可用计算能力往往显得是无限的,并能在任意时候都能获取任意数量的计算能力。

[0039] 可测量的服务:云系统通过利用适于服务类型(例如存储、处理、带宽和活跃用户帐号)的某种抽象程度的计量能力,自动地控制和优化资源效用。可以监测、控制和报告资源使用情况,为服务提供者和消费者双方提供透明度。

[0040] 服务模型如下:

[0041] 软件即服务(SaaS):向消费者提供的能力是使用提供者在云基础架构上运行的应用。可以通过诸如网络浏览器的瘦客户机接口(例如基于网络的电子邮件)从各种客户机设备访问应用。除了有限的特定于用户的应用配置设置外,消费者既不管理也不控制包括网络、服务器、操作系统、存储、乃至单个应用能力等的底层云基础架构。

[0042] 平台即服务(PaaS):向消费者提供的能力是在云基础架构上部署消费者创建或获得的应用,这些应用利用提供者支持的程序设计语言和工具创建。消费者既不管理也不控制包括网络、服务器、操作系统或存储的底层云基础架构,但对其部署的应用具有控制权,对应用托管环境配置可能也具有控制权。

[0043] 基础架构即服务(IaaS):向消费者提供的能力是消费者能够在其中部署并运行包括操作系统和应用的任意软件的处理、存储、网络和其它基础计算资源。消费者既不管理也不控制底层的云基础架构,但是对操作系统、存储和其部署的应用具有控制权,对选择的网络组件(例如主机防火墙)可能具有有限的控制权。

[0044] 部署模型如下:

[0045] 私有云:云基础架构单独为某个组织运行。云基础架构可以由该组织或第三方管理并且可以存在于该组织内部或外部。

[0046] 共同体云:云基础架构被若干组织共享并支持有共同利害关系(例如任务使命、安全要求、政策和合规考虑)的特定共同体。共同体云可以由共同体内的多个组织或第三方管理并且可以存在于该共同体内部或外部。

[0047] 公共云:云基础架构向公众或大型产业群提供并由出售云服务的组织拥有。

[0048] 混合云:云基础架构由两个或更多部署模型的云(私有云、共同体云或公共云)组成,这些云依然是独特的实体,但是通过使数据和应用能够移植的标准化技术或私有技术(例如用于云之间的负载平衡的云突发流量分担技术)绑定在一起。

[0049] 云计算环境是面向服务的,特点集中在无状态性、低耦合性、模块性和语意的互操作性。云计算的核心是包含互连节点网络的基础架构。

[0050] 现在参考图1,其中显示了云计算节点的一个例子。图1显示的云计算节点10仅仅是适合的云计算节点的一个示例,不应对本发明实施例的功能和使用范围带来任何限制。总之,云计算节点10能够被用来实现和/或执行在此所述的任何功能。

[0051] 云计算节点10具有计算机系统/服务器12,其可与众多其它通用或专用计算系统环境或配置一起操作。众所周知,适于与计算机系统/服务器12一起操作的计算系统、环境和/或配置的例子包括但不限于:个人计算机系统、服务器计算机系统、瘦客户机、厚客户

机、手持或膝上设备、多处理器系统、基于微处理器的系统、机顶盒、可编程消费电子产品、网络个人电脑、小型计算机系统、大型计算机系统和包括上述任意系统或设备的分布式云计算技术环境,等等。

[0052] 计算机系统/服务器12可以在由计算机系统执行的计算机系统可执行指令(诸如程序模块)的一般语境下描述。通常,程序模块可以包括执行特定的任务或者实现特定的抽象数据类型的例程、程序、目标程序、组件、逻辑、数据结构等。计算机系统/服务器12可以在通过通信网络链接的远程处理设备执行任务的分布式云计算环境中实施。在分布式云计算环境中,程序模块可以位于包括存储设备的本地或远程计算机系统存储介质上。

[0053] 如图1所示,云计算节点10中的计算机系统/服务器12以通用计算设备的形式表现。计算机系统/服务器12的组件可以包括但不限于:一个或者多个处理器或者处理单元16,系统存储器28,连接不同系统组件(包括系统存储器28和处理器16)的总线18。

[0054] 总线18表示几类总线结构中的一种或多种,包括存储器总线或者存储器控制器,外围总线,图形加速端口,处理器或者使用多种总线结构中的任意总线结构的局域总线。举例来说,这些体系结构包括但不限于工业标准体系结构(ISA)总线,微通道体系结构(MCA)总线,增强型ISA(EISA)总线、视频电子标准协会(VESA)局域总线以及外围组件互连(PCI)总线。

[0055] 计算机系统/服务器12典型地包括多种计算机系统可读介质。这些介质可以是能够被计算机系统/服务器12访问的任意可获得的介质,包括易失性和非易失性介质,可移动的和不可移动的介质。

[0056] 系统存储器28可以包括易失性存储器形式的计算机系统可读介质,例如随机存取存储器(RAM)30和/或高速缓冲存储器32。计算机系统/服务器12可以还包括其它可移动/不可移动的、易失性/非易失性计算机系统存储介质。仅作为举例,存储系统34可以用于读写不可移动的、非易失性磁介质(图1未显示,通常称为“硬盘驱动器”)。尽管图1中未示出,可以提供用于对可移动非易失性磁盘(例如“软盘”)读写的磁盘驱动器,以及对可移动非易失性光盘(例如CD-ROM、DVD-ROM或者其它光介质)读写的光盘驱动器。在这些情况下,每个驱动器可以通过一个或者多个数据介质接口与总线18相连。如下面进一步示出和描述的,存储器28可以包括至少一个程序产品,该程序产品具有一组(例如至少一个)程序模块,这些程序模块被配置以执行本发明各实施例的功能。

[0057] 具有一组(至少一个)程序模块42的程序/实用工具40,可以存储在存储器28中,这样的程序模块42包括但不限于操作系统、一个或者多个应用程序、其它程序模块以及程序数据,这些示例中的每一个或某种组合中可能包括网络环境的实现。程序模块42通常执行本发明所描述的实施例中的功能和/或方法。

[0058] 计算机系统/服务器12也可以与一个或多个外部设备14(例如键盘、指向设备、显示器24等)通信,还可与一个或者多个使得用户能与该计算机系统/服务器12交互的设备通信,和/或与使得该计算机系统/服务器12能与一个或多个其它计算设备进行通信的任何设备(例如网卡,调制解调器等等)通信。这种通信可以通过输入/输出(I/O)接口22进行。并且,计算机系统/服务器12还可以通过网络适配器20与一个或者多个网络(例如局域网(LAN),广域网(WAN)和/或公共网络,例如因特网)通信。如图所示,网络适配器20通过总线18与计算机系统/服务器12的其它组件通信。应当明白,尽管图中未示出,其它硬件和/或软



件组件可以与计算机系统/服务器12一起操作,包括但不限于:微代码、设备驱动器、冗余处理单元、外部磁盘驱动阵列、RAID系统、磁带驱动器以及数据备份存储系统等。

[0059] 现在参考图2,其中显示了示例性的云计算环境50。如图所示,云计算环境50包括云计算消费者使用的本地计算设备可以与其相通信的一个或者多个云计算节点10,本地计算设备例如可以是个人数字助理(PDA)或移动电话54A,台式电脑54B、笔记本电脑54C和/或汽车计算机系统54N。云计算节点10之间可以相互通信。可以在包括但不限于如上所述的私有云、共同体云、公共云或混合云或者它们的组合的一个或者多个网络中将云计算节点10进行物理或虚拟分组(图中未显示)。这样,云的消费者无需在本地计算设备上维护资源就能请求云计算环境50提供的基础架构即服务(IaaS)、平台即服务(PaaS)和/或软件即服务(SaaS)。应当理解,图2显示的各类计算设备54A-N仅仅是示意性的,云计算节点10以及云计算环境50可以与任意类型网络上和/或网络可寻址连接的任意类型的计算设备(例如使用网络浏览器)通信。

[0060] 现在参考图3,其中显示了云计算环境50(图2)提供的一组功能抽象层。首先应当理解,图3所示的组件、层以及功能都仅仅是示意性的,本发明的实施例不限于此。如图3所示,提供下列层和对应功能:

[0061] 硬件和软件层60包括硬件和软件组件。硬件组件的例子包括:主机,例如**IBM® zSeries®**系统;基于RISC(精简指令集计算机)体系结构的服务器,例如**IBM pSeries®**系统;**IBM xSeries®**系统;**IBM BladeCenter®**系统;存储器件;网络和网络组件。软件组件的例子包括:网络应用服务器软件,例如**IBM WebSphere®**应用服务器软件;数据库软件,例如**IBM DB2®**数据库软件。(IBM、zSeries、pSeries、xSeries、BladeCenter、WebSphere以及DB2是国际商业机器公司在全世界各地的注册商标)。

[0062] 虚拟层62提供一个抽象层,该层可以提供下列虚拟实体的例子:虚拟服务器、虚拟存储、虚拟网络(包括虚拟私有网络)、虚拟应用和操作系统,以及虚拟客户端。

[0063] 在一个示例中,管理层64可以提供下述功能:资源供应功能:提供用于在云计算环境中执行任务的计算资源和其它资源的动态获取;计量和定价功能:在云计算环境内对资源的使用进行成本跟踪,并为此提供帐单和发票。在一个例子中,该资源可以包括应用软件许可。安全功能:为云的消费者和任务提供身份认证,为数据和其它资源提供保护。用户门户功能:为消费者和系统管理员提供对云计算环境的访问。服务水平管理功能:提供云计算资源的分配和管理,以满足必需的服务水平。服务水平协议(SLA)计划和履行功能:为根据SLA预测的对云计算资源未来需求提供预先安排和供应。

[0064] 工作负载层66提供云计算环境可能实现的功能的示例。在该层中,可提供的工作负载或功能的示例包括:地图绘制与导航;软件开发及生命周期管理;虚拟教室的教学提供;数据分析处理;交易处理;以及移动桌面。

[0065] 一个或多个实施例有利地提供一种自适应预测系统,以实现即时云资源需求供应。云计算的承诺是每当需要时立即提供计算资源。如上所述,最新虚拟机(VM)供应技术可以在数十分钟内供应VM。这种延迟对于非紧急作业而言可以令人满意,但对于需要在计算期间扩展的作业而言不可接受。新兴的VM流技术可以进一步降低延迟,但仍然受到网络和存储的物理限制。为了真正实现进行中扩展,请求时需要在几秒内准备好新的VM。在一个或

多个实施例中,这通过预测需求和预先准备VM实现。

[0066] 一个或多个实施例提供一种在线时间数据挖掘系统,以便对云VM需求进行建模和预测。一个或多个实施例旨在从VM供应请求流中提取高级特征,并且通知供应系统预先为入站请求准备VM。一个或多个实施例利用云预测成本而不是使用传统的统计优度测量,云预测成本对云的真实成本和约束进行编码以便指导预测算法的训练。在至少某些情况下,这种测量被认为优于传统的回归测量。此外,一个或多个实施例采用一种二级集成(ensemble)方法以便捕获高瞬态需求时间序列的特征。使用来自操作中的云的历史数据的实验结果展示一个或多个实施例能够使供应时间平均减少80%,因此明显改进云服务质量,并且为进行中供应提供可能性。

[0067] 对于业务活动的现代信息技术(IT)解决方案而言,云计算逐渐成为普遍存在的选择。与传统的自主计算数据中心相比,云计算服务的基础架构即服务(IaaS)范例对于业务IT支持而言是更弹性和经济的选择。目前,计算资源客户(公司、科研机构等)每当需要时,可以方便地从云服务提供者(例如IBM、亚马逊和微软)请求计算资源。

[0068] 阻碍云计算的进一步推广的一个问题是VM供应的等待时间。最新VM供应技术平均花费数十分钟来供应VM。因为不能在云中立即准备好计算资源,所以某些IT企业客户仍然针对时间敏感计算首选超能力的传统数据中心,尽管针对硬件、基础架构、维护和能量具有大量的开销成本。客户的这种犹豫不决表明在提供即时资源供应方面仍然存在显著的差距。

[0069] 从硬件和软件技术的角度来看,目前立即并明显减少VM供应时间的希望微乎其微。面向云的硬件体系结构仅可以略微提高供应效率。流VM技术允许客户在VM完全准备好之前预览该VM。此外,因为在足够大的VM部分准备好之前不能立即可用,所以一段时间之后该技术才可以完全符合即时供应需要。

[0070] 从业务的角度来看,一种简单而直接的解决方案是要求所有客户提供未来VM供应请求安排,使得云服务提供者可以事先准备所有VM。但是,由于多种原因,这是不可能的:

[0071] (1) 客户没有义务并且通常不愿意向云服务提供者通知他们将来何时需要更多的资源。

[0072] (2) 客户本身无法知道何时需要计算资源,因为入站工作负载通常不可预测。

[0073] (3) 客户构成始终变化。

[0074] (4) 对于提供请求安排的客户而言,他们的实际请求可能变化。

[0075] 面临这些技术限制和业务约束,一个或多个实施例提供一种实用、可实现并且有效的解决方案,以便通过预测需求并且预先准备VM来提供即时云。

[0076] 云资源预测是一项很有挑战性的任务,这是由于不平衡的需求分布、动态变化的请求和持续的客户流失导致。我们的经验研究表明,针对云资源供应应用传统的时间序列预测技术不能获得可接受的性能。此外,传统的预测技术不能根据实际云环境动态改变预测策略。一个或多个实施例提供一种自适应预测系统,以使云VM需求满足即时供应的要求。一个或多个实施例提供一种实时处理系统,该系统能够随时间而渐进了解和捕获VM需求的最新特征。在一个或多个实施例中,采用一种称为云预测成本(CPC)的新测量,以便指导预测算法的训练过程,并且使用一种二级集成方法在CPC的指导下组合预测结果。针对从IBM Smart Cloud Enterprise (SCE) 平台收集的实际数据进行一系列实验,该平台提供有关数

据过滤和聚合的有用洞察以实现资源供应。一个或多个实施例开发高级数据挖掘技术以实现即时云资源供应。

[0077] 需要强调的是,与在此给出的现有技术相比的实验结果和/或量化改进声明例示一个或多个实施例而非限制;其它实施例可以获得不同的结果和/或不同程度的改进。

[0078] 简要回顾和概述要在此讨论的其它材料:

[0079] 1) 一个或多个实施例在云服务上下文中采用一种称为云预测成本的新测量,以便评估预测结果的质量。云预测成本是一种非对称成本测量,其以不同方式处理预测不足和预测过度的成本。

[0080] 2) 一个或多个实施例提供一种二级集成算法以实现云VM需求预测,其中第一级跨不同算法组合预测结果,第二级通过结合时间序列相关信息来计算更可靠的预测结果。

[0081] 3) 一个或多个实施例提供一种三模块系统,以便根据请求历史预测多类型VM需求。

[0082] 4) 提供针对实际数据的一系列实验的结果,它们展示云预测成本能够指导预测过程和/或根据本发明的一个或多个方面的一个或多个示例性系统能够明显减少平均供应时间,同时还成功控制资源的工作负载。

[0083] 云资源供应

[0084] 在IBM Smart Cloud Enterprise (SCE) 平台中,每个VM请求包括21个特性,例如客户ID、VM类型、请求开始时间等。出于示例性目的,本说明书专注于针对特性VM类型和请求时间的聚合时间序列。

[0085] 可以按照下面描述通常在VM需求预测中提到的挑战:

[0086] 1) 给定请求记录流S,需要一种高效并且适当的机制以便适当地从流中提取高级聚合信息,并且将该信息变换为多个时间序列 $T = (T_1, T_2, \dots, T_k)$ ,每个映像类型一个时间序列。

[0087] 2) 获得时间序列集合T,需要一种模型选择和训练机制以便准确、定期并及时地生成对应的预测模型P。

[0088] 3) 获得最新预测模型集合P,需要一种有效并且准确的时间序列预测器以便根据历史请求记录,预测时间 $t_{n+1}$ 的需求量。

[0089] 图4示出VM需求随时间的示例性时间序列。顶部时间序列402示出所有VM类型的需求,而底部三个时间序列404、406、408示出三个需求最高的映像类型的时间序列。图4中的数据被规范化。

[0090] 总需求的时间序列402反映清晰的每周需求变化周期,并且预测算法可以很容易地捕获其特征(多个)。但是,对总需求进行建模和预测未给供应时间的减少带来任何帮助,因为所有供应与VM类型关联。当按照VM类型分解请求时,具有不同VM类型的时间序列(例如,图1中的底部3个时间序列404、406、408)变得非常不规则。

[0091] 通过实验研究,我们发现在其中希望执行VM请求预测的一个或多个情况下通常遇到以下困难:

[0092] 1) 不同VM类型的需求分布严重不平衡。此外,分布随时间发生巨大变化。

[0093] 2) 当存在动态变化的客户群体时,从长期的角度来看,某些时间序列包含可以误导预测算法的随机因素。

[0094] 3) 存在各种不同类型的时间序列,它们具有完全不同的特征。没有一种现有预测算法能够很好地针对所有这些时间序列执行。

[0095] 因此,请求预测任务并不像直接的时间序列预测任务那样简单。

#### [0096] 系统框架

[0097] 图5示出根据本发明的一个方面的示例性系统框架。为了减少供应时间,图5的示例性实施例提供一种实时预测系统500,该系统能够随时间而渐进了解和捕获供应需求的最新特征。如图所示,可以将该系统分为三个模块,其中每个模块执行在此描述的一种对应功能。

[0098] 原始数据过滤与聚合模块501针对后续模块执行预处理工作。该模块具有两个任务:(1)从源自请求历史数据库510的原始数据流中滤除不必要的信息(包括不相关的特性和记录);以及(2)提取过滤后的数据的高级特征并且将它们变换为时间序列。

[0099] 模型生成模块502负责构建单独预测器的模型。它根据原始数据预处理模块501提供的最新数据,定期选择和训练每个预测器的最合适的模型。完成工作之后,将训练后的模型的新参数存储到特定文件中,并且通知需求预测模块503。

[0100] 需求预测模块(例如可以通过集成预测器503以及方框502实现)根据存储在模型文件中的信息,重构预测模型。它然后利用回归集成和相关集成这一两阶段集成机制来预测未来需求。请参见下面的20、522的讨论以及其他位置的预测集成的讨论。

[0101] 具体地说,消费者512向请求处理机514发送请求,如516所示;将所述请求添加到请求历史数据库510,如518所示。然后为模块501提供原始数据,如上所述。然后将由模块501预处理的数据提供给模块502和集成器(ensemble) 520,集成器520还从模块502获得模型参数。集成器520包括用于不同类型VM的适当预测算法。组合器522然后组合预测器,如在预测集成的讨论中解释的那样。如524所示和包括结合图19在其他位置讨论的,根据对预测的分析,将适当数量的VM放在池526中;请求处理机514从池526中取回VM,如528所示,并且处理机514将VM提供给客户512,如530所示。

#### [0102] 云预测成本

[0103] 传统的回归成本测量,例如平均误差(MAE)、最小平方误差(LSE)和平均绝对百分比误差(MAPE),全部是对称测量并且仅专注于回归结果的绝对准确度。在云需求预测中,预测过度和预测不足的结果完全不同。预测过度对客户没有负面影响,并且仅导致浪费云服务提供者的资源,而预测不足节省资源成本,但降低客户的服务质量。此外,传统的回归成本测量不能指导预测器在不同系统状态和服务水平协议(SLA)要求下自由改变预测策略。

[0104] 考虑云需求预测的独特性,一个或多个实施例采用云预测成本(CPC)这一非对称并且异构的成本测量,以便指导模型选择。CPC优于以不同方式处理预测过度和预测不足的传统成本敏感测量,并且还考虑两个不同类型的成本:客户服务质量损失的成本和闲置资源的成本。

[0105] 假设对于某个类型的VM,在时间 $t$ 处的实际请求(需求)数量是 $v(t)$ ,在时间 $t$ 处的供应数量是 $\hat{v}(t)$ ,其中 $\hat{v}(t)$ (预测的需求)是历史值 $v(t-1)$ 、 $\dots$ 、 $v(t-i)$ 的函数。定义两个函数 $T(v(t), \hat{v}(t))$ 和 $R(v(t), \hat{v}(t))$ ,以便分别表示服务质量的成本和闲置资源的成本。成本函数针对每个云而有所变化。一个或多个实施例可以与满足以下两个属性的任何成本函数

协同工作：

[0106] 1) 非负：对于所有  $v(t)$  和  $\hat{v}(t)$ ,  $T(v(t), \hat{v}(t)) \geq 0$  并且  $R(v(t), \hat{v}(t)) \geq 0$ 。

[0107] 2) 单调：如果  $v_1(t) - \hat{v}_1(t) \geq v_2(t) - \hat{v}_2(t)$ , 则  $T(v_1(t), \hat{v}_1(t)) \geq T(v_2(t), \hat{v}_2(t))$ 。同样, 如果  $\hat{v}_1(t) - v_1(t) \geq \hat{v}_2(t) - v_2(t)$ , 则  $R(v_1(t), \hat{v}_1(t)) \geq R(v_2(t), \hat{v}_2(t))$ 。

[0108] 服务质量的成本：VM供应时间是重要的服务质量指标。当VM供应请求到达时, 如果已经预先准备同一类型的VM, 则系统可以简单地将所有权转移给客户并且改变相关记录。这最多只需几秒。将该短供应时间记为  $T_{hit}$ , 这类似于系统高速缓存情形中的高速缓存“命中”。如果没有准备好的VM来满足客户的请求, 则必须在进行中 (on-the-fly) 准备VM, 这在此称为“未命中”。将等待时间记为  $T_{miss}$ 。一般而言, 供应时间  $T_{miss} \gg T_{hit}$ 。在当前实际云服务系统中, 单独对于机械操作而言, 进行中供应时间  $T_{miss}$  是数十分钟。对于企业云, 如果涉及任何手动过程 (多个), 则用于软件完整性和安全性的质量保证过程甚至可以需要数天。

[0109] 可以将一种简单形式的  $T(v(t), \hat{v}(t))$  建模为所有VM的总供应时间。在预测过度的情况下, 即  $v(t) \leq \hat{v}(t)$ , 可以立即满足所有供应请求。在预测不足的情况下, 即  $v(t) > \hat{v}(t)$ , 需要在进行中准备所有预测不足的VM部分。因此, VM供应时间的成本是：

$$[0110] \quad T(v(t), \hat{v}(t)) = \min(v(t), \hat{v}(t))T_{hit} + \max(0, v(t) - \hat{v}(t))T_{miss}, \quad (1)$$

[0111] 其中  $T_{miss}$  和  $T_{hit}$  是常量。根据该定义, 供应时间越长, 质量越低。

[0112] 还存在更复杂形式的  $T$  函数。例如, 公共SLA指定供应时间的惩罚阈值。如果在不违反阈值的情况下满足请求, 则没有针对云提供者的惩罚。在这些情况下, 服务质量成本函数在SLA阈值下针对供应时间具有0值。

[0113] 闲置资源的成本：这是不可计费的资源成本, 包括磁盘空间、网络带宽、电力成本以及人工成本等。为简单起见, 在此使用  $R_{vm}$  表示单个VM的所有资源的成本。出于示例性目的, 假设单位时间内每个VM的闲置资源成本相同, 而无论虚拟机映像的类型为何。为使实例简单, 还假设所有VM具有相同的大小。将闲置资源的成本函数定义为：

$$[0114] \quad R(v(t), \hat{v}(t)) = \max(0, \hat{v}(t) - v(t))R_{vm}, \quad (2)$$

[0115] 其中  $R_{vm}$  是常量。

[0116] 还可以使用更复杂形式的  $R(v(t), \hat{v}(t))$ 。例如, 在实践中, 始终为VM供应提供固定量的资源  $R_{fix}$ , 即固定资源。如果过度预测的值低于此类阈值, 则资源成本仍然为0。等式 (3) 示出该成本函数的形式：

$$[0117] \quad R(v(t), \hat{v}(t)) = \begin{cases} 0 & \text{如果 } \hat{v}(t) < \min(R_{fix}, v(t)), \\ (\hat{v}(t) - R_{fix}) R_{vm} & \text{如果 } v(t) < R_{fix} < \hat{v}(t), \\ (\hat{v}(t) - v(t)) R_{vm} & \text{如果 } R_{fix} \leq v(t) < \hat{v}(t). \end{cases}$$

(3)

[0118] 组合等式 (1) 和 (2), 通过等式 (4) 量化预测质量的简单成本函数。图6的表中列出所有相关符号。

$$[0119] \quad C = \beta T(v(t), \hat{v}(t)) + (1 - \beta) \times R(v, \hat{v}(t))$$

$$= \begin{cases} \beta v(t) T_{hit} + (1 - \beta) (\hat{v}(t) - v(t)) R_{vm}, & \text{如果 } v(t) < \hat{v}(t) \\ \beta (\hat{v}(t) T_{hit} + (v(t) - \hat{v}(t)) T_{miss}) \circ & \text{如果 } v(t) \geq \hat{v}(t) \end{cases}$$

(4)

[0121] 云资源需求的最佳预测器是在预测值  $\hat{v}(t)$  和实际值  $v(t)$  之间获得最低总成本的预测器。

[0122] 定期模型更新: “训练一次, 永远预测” 的理念被认为不适合于云VM需求预测。由于云服务的独特特征, VM类型的时间序列具有高度瞬态性。为了捕获时间序列的最新特征, 一个或多个实施例根据新请求历史, 定期更新预测模型。对于每次更新, 一个或多个实施例利用网格搜索策略和十倍验证方法, 以便针对每个VM类型选取每个预测模型的最佳参数组合。

#### [0123] 需求预测

[0124] 在一个或多个实施例中, 需求预测模块的核心是二级集成技术。第一级是基于回归的集成, 其组合同一VM类型的不同预测模型的结果。第二级集成考虑不同VM类型之间的关系, 并且使用它们的相关性以便帮助提高预测的可靠性。

[0125] 基本预测方法: 一个或多个实施例采用一组不同预测技术作为基本预测器。图7的表中列出预测技术。

[0126] 预测集成: 为了以合适的方式处理预测任务, 一个或多个实施例采用一种预测集成方法来组合个体预测技术的能力。

[0127] 因为分类原理和预测问题之间的差异, 不可能直接使用基于分类的集成方法的理论。对于分类, 目标是通过多数表决获得更准确的集成分类器。集成分类器的性能通常优于基本分类器。对于预测, 目标是针对不同类型的时间序列, 使集成预测器更可靠并且使其精度接近于最佳预测器。

[0128] 一个或多个实施例针对需求预测采用加权式线性组合集成策略。假设预测器  $p \in P$  的预测值是  $\hat{v}_p$ , 并且其在时间  $t$  处的对应权重是  $w_p^{(0)}$ , 则某个VM类型在时间  $t$  处的预测值是:

$$[0129] \quad \hat{v}^{(t)} = \sum_p w_p^{(t)} \hat{v}_p, \text{ 遵循 } \sum_p w_p^{(t)} = 1. \quad (5)$$

[0130] 初始地 ( $t=0$ ), 因此对于每个预测器  $p$ ,  $w_p^{(0)} = \frac{1}{|P|}$ , 这样预测器对于组合后的预测

结果具有相同的贡献。

[0131] 基于预测的集成的权重更新策略也不同于传统的基于分类的策略。在分类情形中, 结果只能为“正确”或“不正确”。需要的只是增加那些正确分类的分类器的权重以实现权重更新。但是, 在预测情形中, 结果是连续值, 并且预测器的权重将直接影响集成结果。因此, 应该谨慎量化更新策略。

[0132] 一个或多个实施例利用每个预测值  $\hat{v}_p(t)$  和实际值  $v(t)$  之间的差。为了更新权重, 根据以下等式计算预测器  $p$  在时间  $t-1$  处导致的相对误差  $e_p^{(t-1)}$ :

$$[0133] \quad e_p^{(t)} = \frac{\sum_p c_p^{(t-1)}}{c^{(t-1)}} w_p^{(t-1)}, \quad (6)$$

[0134] 其中  $c_p^{(t-1)}$  是预测成本并且可以由任何种类的成本函数给出, 这些成本函数如MAE、LSE、MAPE和CPC (如在上面云预测成本部分中定义的)。

[0135] 注意, 在一个或多个实施例中, 相对误差不能用作预测器的新权重, 因为它们未被规范化。因为最终预测值是个体预测器的所有结果的线性组合, 所以使用等式 (7) 使约束  $\sum_p w_p^{(t)} = 1$  成立:

$$[0136] \quad w_p^{(t)} = \frac{e_p^{(t)}}{\sum_p e_p^{(t)}}. \quad (7)$$

[0137] 很容易证明, 通过此权重更新策略, 保证增加最佳预测器在每个时间处的权重。假设  $\bar{c}$  是所有预测器在时间  $t-1$  处的平均成本。

[0138] 根据等式 (6):

$$[0139] \quad \frac{\sum_p c_p^{(t-1)}}{c^{(t-1)}} = \frac{|P| \sum_p c_p^{(t-1)}}{|P| c^{(t-1)}} \begin{cases} > \frac{|P| \sum_p c_p^{(t-1)}}{\sum_p c_p^{(t-1)}} & \text{如果 } c^{(t-1)} < \bar{c}, \\ < \frac{|P| \sum_p c_p^{(t-1)}}{\sum_p c_p^{(t-1)}} & \text{如果 } c^{(t-1)} > \bar{c}. \end{cases}$$

[0140] 如果预测器的成本小于平均成本, 则  $c^{(t-1)}$  的增加率大于  $|P|$ 。相反, 如果预测器的成本大于平均成本, 则  $c^{(t-1)}$  的增加率小于  $|P|$ 。因为规范化保留权重的顺序, 所以那些具有优于平均性能的预测器在下一轮变得更重要 (权重增加)。

[0141] 相关性集成:在实际云服务情形中,某些VM类型的需求可能看似非常不规则。当使用不规则值作为训练数据时,它们将误导预测算法。为了减少“噪声”数据的干扰,一个或多个实施例还利用时间序列之间的相关性信息以便帮助预测。例如,具有不同平台(例如,32位和64位)的同一软件的需求时间序列高度相关。

[0142] 一个或多个实施例使用时间序列的相关矩阵,以便帮助对预测进行后处理。假设  $\Sigma(t)$  是最常见VM在时间  $t$  处的协方差矩阵(例如,设计中的前12个时间序列),  $\text{cov}_{ij}^{(t)}$  表示资源类型  $i$  及其第  $j$  个相关资源之间的协方差。通过考虑强相关的时间序列(相关性大于0.8)的正面影响,时间序列  $i$  在时间  $t$  处的预测值  $\hat{u}_i^{(t)}$  现在变成:

$$[0143] \quad \hat{u}_i^{(t)} = \frac{\sum_{j=1}^k \text{cov}_{ij}^{(t-1)} s_{ij} \hat{u}_j^{(t)}}{\sum_{j=1}^k s_{ij} \text{cov}_{ij}}, \quad (8)$$

[0144] 其中  $s_{ij} = \bar{t}_i / \bar{t}_j$  表示两个时间序列之间的规模差异,  $k$  是强相关时间序列的数量。在一个或多个示例性实施例中,仅考虑正相关的时间序列。

[0145] 预留控制器:预留控制器是一种模块,其用于一个或多个实施例中以便与VM供应系统通信。它预留未使用的VM,并且仅当被请求类型的所有预留都用完时,才通知VM供应系统准备新VM。预留控制器提供一种良好的缓冲机制,其有效地减少预测过度的浪费。

#### [0146] 系统评估

[0147] 在此描述的实验评估基于从当前云服务平台获得的实际历史VM请求数据。历史需求数据包含具有多于100个不同类型的数万个VM请求。数据的时间跨度超过3个月。

[0148] 评估的目标是回答以下问题:

[0149] ● 什么是最佳需求预测策略?

[0150] ● 集成方法是否比单独的预测技术更可靠并且有效?

[0151] ● 一个或多个实施例可以在什么程度上减少供应时间,并且一个或多个实施例可以导致多少闲置资源?

[0152] ● 对于云需求预测而言,云预测成本是否实用并且灵活?

#### [0153] 数据过滤与聚合

[0154] 数据预处理是需求预测的第一步—参见上面的模块501的讨论。数据流中的原始请求记录不能直接用于需求预测,这有两个原因:

[0155] (1) 预测算法不能直接针对低级表示构建模型。

[0156] (2) 原始数据包含无用请求记录或可以导致预测不精确的不相关特性。

[0157] 请求记录的21个特性并非全部有用。在一个或多个实施例中,仅考虑VM类型,其示出客户请求的VM类型;以及请求时间戳,其指示客户发送请求的时间。

[0158] 时间序列聚合粒度:经由VM类型和请求时间戳,通过聚合请求记录生成时间序列集合。对于请求时间戳,可以使用不同粒度进行聚合。图8示出同一VM类型在数周、数天和数小时内的时间序列;即,具有不同粒度的时间序列(规范化后的数据)。顶部时间序列802每周聚合,中间时间序列804每天聚合,底部时间序列806每小时聚合。

[0159] 很容易知道,粒度越粗糙,每个时间段的需求越大。如果以粗糙粒度(例如,每周)



聚合时间序列,则需要系统准备过多的VM,并且大多数准备好的VM将空闲很长时间。此外,与更细粒度相比,即使较小部分的预测过度也将导致更多的闲置资源。

[0160] 相反,以细粒度(例如,每小时)聚合将使时间序列缺乏统计意义并且难以预测。

[0161] 图9的表示出变异系数(CV)、偏度和峰度下的VM需求时间序列特征,这三个统计量用于测量时间序列的形状。尽管认为本领域技术人员将熟悉CV、偏度和峰度,然而仍参考David Moore和George McCabe的Introduction to the Practice of Statistics(统计学实践导论),W.H.Freeman and Company,其所有内容在此明确引入作为参考以用于所有目的。较高的CV值指示较大的波动性,较高的偏度指示较强的非对称性,较高的峰度表示更多变异是罕见极限偏差的结果。相比之下,每小时聚合的时间序列在所有三个度量中具有最大值,从而指示小时粒度也不适合于聚合。因此,在随后的实验中,每天执行聚合。

[0162] VM类型选择:图10示出时间-类型-计数视图中的请求分布(规范化后的数据);也就是说,VM类型随时间的分布。图10示出两个相关特征;即,VM请求分布严重不均匀,因为少量VM类型主导了分布。图11A和11B提供CDF图和请求频率图(规范化后的数据);具体地说,图11A示出对应的累积分布函数(CDF),图11B根据其请求频率对VM进行排序。CDF示出VM请求服从80/20规则;即,超过80%的请求集中于不到20%的VM类型。在排序图中,观察到在第12和第13类型之间具有拐点,该拐点明确地将类型分为常见组和罕见组。针对这些少数时间序列的测量(例如变异系数、偏度和峰度)还示出高于针对多数时间序列的测量的值,从而指示这些少数类型的时间序列并不足够规则以便建模和预测。但是,注意,这不意味着不能针对罕见VM类型采取操作以便缩短供应时间。

[0163] 根据上面研究的结果,数据过滤与聚合模块旨在定期检查排序变化,并且仅过滤前十二个常见类型以便生成它们对应的时间序列。对于那些少数VM类型,始终可使用单纯策略,例如针对每个罕见VM类型准备固定数量的VM。因为这些VM类型罕见,所以预测误差成本微不足道。

[0164] 预测准确度

[0165] 将不同基本预测器的预测性能与根据本发明一个方面的集成预测器的示例性实施例相比。除了本发明的成本函数CPC之外,还使用MAE、MSE和MAPE测量这些预测器的准确度。对于CPC的参数,设置 $\beta=0.5$ (以同样方式处理供应时间的成本和闲置VM的成本), $T_{\text{miss}}=1040$ (如果未预先准备一个VM,则客户需要等待超过10分钟), $T_{\text{hit}}=5$ (如果准备一个VM,则客户可以在5秒内获得该VM), $R_{\text{vm}}=400$ (一个闲置VM的成本)。

[0166] 为了评估预测器的可靠性,选择前十二个(在图11B中的拐点之前)最常见VM类型的时间序列以进行实验。将所有时间序列水平(沿着时间轴)划分为两个集合,最近30天的时间序列用作测试数据集,而剩余请求用作训练集。对于基本预测器,使用网格搜索策略寻找最佳参数组合。

[0167] 图13的表示出所有预测器在所有时间序列上的准确度。为了避免混乱,仅列出前三个时间序列的详细信息;即,Red Hat Enterprise Linux 5.532位版本1302、Red Hat Enterprise Linux 5.564位版本1304和SUSE Linux Enterprise Server 1306,以及所有时间序列的平均CPC 1308。可以看到,不同VM类型的最佳预测器是不同的。例如,GEP针对第1VM类型1302最佳地执行;而ANN和GEP分别针对第2VM类型1304和第3VM类型1306获得良好结果。此外,一个VM类型的优胜预测器还可以针对其它VM类型较差地执行。例如,ANN针对第1VM类

型1302获得较差性能。

[0168] 在一个或多个实施例中,对于集成预测器,尽管它没有针对任何单个VM类型最佳地执行,但它非常可靠,因为其性能始终接近于针对所有类型的优胜预测器。平均CPC示出集成预测器具有最佳平均性能,从而指示其自适应能力。

[0169] 图14A-14C示出三个最常见VM类型的实际时间序列和所有预测器的对应预测结果。在该图中,集成预测器可以始终标识时间序列的最佳预测器,并且快速汇聚到该预测器。因为在云供应情形中预测不足差于预测过度,所以很少对需求预测不足的预测器被视为优于其输出始终接近于实际需求的预测器。还可以注意,尽管MA和SVM在三个VM类型的任何一个中没有最佳性能,但它们也可以根据其权重对集成预测器做出贡献。

[0170] 具体地说,在图14A-14C中,提供针对三个最常见时间序列的测试数据的预测结果(规范化后的数据)。从上到下,时间序列是:实际时间序列(实际);MA预测的时间序列;AR预测的时间序列;ANN预测的时间序列;GEP预测的时间序列;SVM预测的时间序列;以及根据本发明的一个或多个方面的集成(ENS)技术预测的时间序列。

[0171] 详细成本

[0172] 供应时间减少:性能评估中的一个重要准则是可以节省多少供应时间。一个或多个实施例根据等式(9)计算预测器获得的节省时间部分,其中根据等式(1)计算T函数:

$$[0173] \quad P_{save} = \frac{\sum_t (v(t)T_{miss} - T(v(t), \hat{v}(t)))}{\sum_t v(t)T_{miss}} \quad (9)$$

[0174] 图12A示出针对每个预测器节省的时间部分。可以看到,大多数预测器可以明显使供应时间减少60%以上。但是,当跨时间序列存在较大变化时,预测器实现的节省时间对于不同VM类型而言并不稳定。平均起来,集成预测器由于其强大的自适应能力而最佳地执行。

[0175] 闲置资源:闲置资源的成本是预测质量的另一个评估准则。诚然,始终预测过度的预测器可以节省大量供应时间,但这种预测器也将浪费大量资源。图12B示出每个预测器导致的闲置资源量。平均起来,最佳资源节省者是SVM,但其在时间减少方面的性能最差。GEP在时间减少方面获得良好性能,但其浪费的资源是根据本发明的一个或多个实施例的技术的两倍之多。

[0176] 不同成本测量的比较

[0177] 图15的表示出各种成本测量指导的集成预测器的时间减少和闲置资源。CPC在供应时间减少方面明显优于其它三个成本测量。还应该注意,CPC具有最多的闲置资源。可以通过这些成本测量的基本概念很好地解释这种现象。MAE、MSE和MAPE都是对称成本测量,它们指导集成预测器以同样方式对不足预测器和过度预测器进行加权。尽管CPC为不足预测器提供的惩罚多于过度预测器,但集成预测器始终首选为过度预测器提供更大权重,这导致更多的时间减少,并且还更可能浪费资源。如在云服务情形中,客户服务质量更重要,并且针对合理数量闲置资源的成本减少供应时间是值得的。

[0178] 预留控制器

[0179] 图16的表示出可以通过结合预留控制器实现的供应时间减少的比率。对于所有时间序列,预留控制器进一步改进平均供应时间的减少,从83.79%改进到94.22%。此外,在预留控制器的帮助下,可以在随后数天内使用以前准备的预测过度的VM部分,而无需再次

经历供应过程。

#### [0180] $\beta$ 效应

[0181] 如前所述,参数 $\beta$ 用于调整服务质量成本和闲置资源成本之间的重要性。更高的 $\beta$ 导致集成预测器为过度预测器分配更大权重。图17中的时间减少/闲置资源曲线示出 $\beta$ 如何影响这两个因素之间的平衡。如预期的,当 $\beta$ 增加时,时间减少的比率和闲置资源都增加。当 $\beta$ 等于0时,平均供应时间减少的比率相当有限。这是因为服务质量损失的成本被完全忽略,并且集成方法非常悲观地预测需求。当 $\beta$ 等于1时,闲置资源的成本巨大,但平均供应时间减少最高。

[0182] 该实验结果确认 $\beta$ 是一种用于设置服务质量损失和闲置资源成本之间的权衡的有效参数。在实践中,可以在需要时动态调整 $\beta$ 。

#### [0183] 复杂成本函数的效应

[0184] 如在有关云预测成本的部分中所述,根据一个或多个实施例的框架很灵活,并且可以使用不同成本函数来指导预测过程。考虑使用在等式(3)中定义的成本函数如何影响预测的需求时间序列。等式(3)中的唯一参数是 $R_{fix}$ ,其指示固定资源量。图18示出由不同 $R_{fix}$ 值生成的三个预测的需求时间序列。当 $R_{fix}$ 增加时,预测的需求倾向于更乐观而不是悲观。实验结果表明,当 $R_{fix}$ 增加到200时,平均供应时间减少93.94%,但闲置资源量变得巨大。

#### [0185] 其它方面的重述和讨论

[0186] 值得注意的是,先前针对计算机系统数据的数据挖掘工作通常需要很好地定义常规行为的模式,并且分析基于相对静态和稳定的环境。另一方面,本发明的一个或多个实施例应对不稳定客户构成和多变需求的存在,使得现有技术方法不能直接用于请求预测。此外,先前的时间序列预测技术通常不能处理本发明的一个或多个实施例的某些应用中要求的所有预测任务,这是由于瞬态环境所致,而一个或多个实施例采用基于回归的集成方法以组合这些技术的预测结果。此外,应该注意,一个或多个实施例预测整个云的VM需求,而不是预测VM中的资源使用。

[0187] 按需计算的理念导致云计算的普及。但是,当前云计算尚未真正实现提供即时弹性的目标。数十分钟的机械处理和数小时的手动操作为进行中虚拟机供应带来严重的障碍。一个或多个实施例采用数据挖掘实现全新类型的服务。

[0188] 一个或多个实施例提供用于减少客户等待VM供应的时间的技术。各个方面包括即时资源供应系统以便处理云VM需求预测。由于云服务的独特成本结构,一个或多个实施例利用云预测成本来测量预测模型的性能。此外,一个或多个实例采用一种二级集成方法以使预测结果可靠并容忍尖峰。一系列实验展示根据本发明一个方面的预测系统的一个或多个示例性实施例可以有效地减少客户等待VM的时间,同时不会导致大量闲置资源。

[0189] 图19示出示例性预供应决策步骤。将有关什么客户请求什么VM的历史数据存储在供应日志数据库1902中。此外,新客户可以提供有关他们何时将工作负载从现有环境迁移到云服务提供者提供的云环境的信息。这种有关未来迁移的计划数据可以存储在1920中。趋势分析模块1904使用适当的技术以便例如确定给定类型VM每天被请求十次,增长率为每单位时间如此之多的VM。结构分析模块1906确定与给定应用关联的映像;例如,购物网站应用可以是三层结构,Web服务器在前面,其与客户对接;应用服务器在Web服务器后面以便执

行业务逻辑;以及一个或多个数据库。每层可以具有多个实例。客户简档建立模块1908允许按照技术和类型对客户进行分类;例如,客户可以使用WINDOWS系统,并且可以测试云环境或者可以致力于云环境。模块1904、1906、1908均从供应日志1902获得输入。模块1908还从迁移计划1920获得输入。迁移需求分析模块1910使用客户的迁移计划1920,预测不可通过客户的历史数据预测的未来需求。这主要发生在新客户以及计划将大量工作负载迁移到云中的客户处。

[0190] 需求预测模块1912聚合所有输入分析,以便预测未来时间段内需要的映像。需求预测模块1912聚合元件1904、1906、1908和1910的所有分析,并且利用二级时间序列集成机制来预测未来需求。第一集成级别采用一组不同时间序列预测器,例如SVM(支持向量机)、人工神经网络和ARMA(自回归移动平均),然后组合它们的输出;第二集成级别使用不同映像类型的相关信息以便获得最终预测的需求。预供应分析模块1914根据预测的需求和迁移计划1920,确定预供应什么VM以及当预测的需求变化时针对已经供应的VM执行什么操作,并且发出预供应订单1918以实现所述操作;可以将该订单发送到合适的预供应引擎。

[0191] 给出到目前为止的讨论,应该理解,一般地说,根据本发明的一个方面的示例性方法包括以下步骤:从云计算环境的虚拟机供应请求流中提取高级特征;例如,参见图5中的步骤和元件516、514、518、510和501的讨论。还包括以下步骤:针对多个不同类型的虚拟机生成多个预测模型。例如,参见图5中的元件502和503的讨论。进一步的步骤包括将回归-相关集成应用于预测模型中的适当模型,以便预测云计算环境的未来虚拟机需求。例如,参见图5中的元件520、522以及图19中的方框1912的讨论。

[0192] 在某些情况下,进一步的步骤包括根据预测的未来虚拟机需求,在云计算环境中预供应多个虚拟机。例如,参见图5中的步骤或元件524、526、528、514、530以及图19中的方框1914、1918的讨论。

[0193] 在一个或多个实施例中,提取501包括过滤不相关的特性和记录,并将高级特征变换为时间序列。在某些实例中,所述过滤将滤除虚拟机类型和请求时间之外的特征。

[0194] 在某些情况下,基于以不同方式处理预测不足成本和预测过度成本的非对称成本测量来执行所述生成步骤;在一种优选但非限制性方法中,基于云预测成本执行所述生成步骤。云预测成本在此被定义为等式(4)的成本函数。

[0195] 如上所述,在某些实施例中,当虚拟机供应请求流在时间上向前移动时,重复所述提取和生成步骤。

[0196] 在一个或多个实例中,所述应用步骤包括:经由加权后的线性组合来组合多个所述预测模型以便获得中间预测;以及经由时间序列中的正相关序列之间的相关性对所述中间预测进行后处理,以便获得所预测的未来虚拟机需求。参考上面的预测集成和相关性集成的讨论。

[0197] 如结合预留控制器讨论的,在某些情况下,额外步骤包括根据所预测的未来虚拟机需求,在所述云计算环境中预留预供应的虚拟机中的空闲虚拟机。

[0198] 示例性系统和制造品细节

[0199] 所属技术领域的技术人员知道,本发明的各个方面可以实现为系统、方法或计算机程序产品。因此,本发明的各个方面可以具体实现为以下形式,即:完全的硬件实施方式、完全的软件实施方式(包括固件、驻留软件、微代码等),或硬件和软件方面结合的实施方案

式,这里可以统称为“电路”、“模块”或“系统”。此外,本发明的各个方面还可以实现为在一个或多个计算机可读介质中的计算机程序产品的形式,该计算机可读介质中包含计算机可读的程序代码。

[0200] 本发明的一个或多个实施例或其元素可以以装置的形式实现,所述装置包括存储器和至少一个处理器,所述至少一个处理器耦合到所述存储器并可操作以执行示例性方法步骤。

[0201] 一个或多个实施例可以使用在通用计算机或工作站上运行的软件。参考图1,这种实现例如可以采用处理器16、存储器28,以及到显示器24和外部设备(多个)14(例如键盘、指点设备等)的输入/输出接口22。术语“处理器”如在此使用的,旨在包括任何处理设备,例如包括CPU(中央处理单元)和/或其它形式处理电路的处理设备。进一步,术语“处理器”可以指多个单独的处理器。术语“存储器”旨在包括与处理器或CPU关联的存储器,例如RAM(随机存取存储器)30、ROM(只读存储器)、固定存储器件(例如,硬盘驱动器34)、可移动存储器件(例如,软盘)、闪存等。此外,词组“输入/输出接口”如在此使用的,旨在构想例如到以下各项的接口:一个或多个用于将数据输入到处理单元的机构(例如,鼠标),以及一个或多个用于提供与处理单元关联的结果的机构(例如,打印机)。处理器16、存储器28和输入/输出接口22例如可以经由总线18互连,作为数据处理单元12的一部分。还可以例如经由总线18为网络接口20(例如网卡,其可以用于与计算机网络对接)以及介质接口(例如软盘或CD-ROM驱动器,其可以用于与合适的介质对接)提供合适的互连。

[0202] 因此,如在此描述的,包括用于执行本发明方法的指令或代码的计算机软件可以被存储在一个或多个关联的存储器件(例如,ROM、固定或可移动存储器)中,并且当准备使用时,被部分或全部加载(例如,加载到RAM中)并由CPU实现。这种软件可以包括但不限于固件、驻留软件、微代码等。

[0203] 适合于存储和/或执行程序代码的数据处理系统将包括至少一个通过系统总线18直接或间接耦合到存储元件28的处理器16。存储元件可以包括在程序代码的实际执行期间采用的本地存储器、大容量存储装置以及提供至少某些程序代码的临时存储以减少必须在执行期间从大容量存储装置取回代码的次數的高速缓冲存储器32。

[0204] 输入/输出或I/O设备(包括但不限于键盘、显示器、指点设备等)可以直接或通过中间I/O控制器与系统相连。

[0205] 网络适配器20也可以被连接到系统以使数据处理系统能够通过中间专用或公共网络变得与其它数据处理系统或远程打印机或存储器件相连。调制解调器、电缆调制解调器和以太网卡只是当前可用的网络适配器类型中的少数几种。

[0206] 如在此(包括权利要求)使用的,“服务器”包括运行服务器程序的物理数据处理系统(例如,如图1中所示的系统12)。将理解,这种物理服务器可以包括也可以不包括显示器和键盘。

[0207] 如上所述,本发明的各个方面可以实现为在一个或多个计算机可读介质中的计算机程序产品的形式,该计算机可读介质中包含计算机可读的程序代码。可以采用一个或多个计算机可读介质的任意组合。计算机可读介质可以是计算机可读信号介质或者计算机可读存储介质。计算机可读存储介质例如可以是一但不限于一电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者上述的任意合适的组合。计算机可读存储介质的更具体的例

子(非穷举的列表)包括:具有一个或多个导线的电连接、便携式计算机盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。在本文件中,计算机可读存储介质可以是任何包含或存储程序的有形介质,该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。

[0208] 计算机可读的信号介质可以包括例如在基带中或者作为载波一部分传播的数据信号,其中承载了计算机可读的程序代码。这种传播的数据信号可以采用多种形式,包括但不限于一电磁信号、光信号或上述的任意合适的组合。计算机可读的信号介质可以是计算机可读存储介质以外的任何计算机可读介质,该计算机可读介质可以发送、传播或者传输用于由指令执行系统、装置或者器件使用或者与其结合使用的程序。

[0209] 计算机可读介质上包含的程序代码可以用任何适当的介质传输,包括但不限于一无线、有线、光缆、RF等等,或者上述的任意合适的组合。

[0210] 可以以一种或多种程序设计语言的任意组合来编写用于执行本发明的各个方面的操作的计算机程序代码,所述程序设计语言包括面向对象的程序设计语言—诸如Java、Smalltalk、C++等,还包括常规的过程式程序设计语言—诸如“C”语言或类似的设计语言。程序代码可以完全地在用户计算机上执行、部分地在用户计算机上执行、作为一个独立的软件包执行、部分在用户计算机上部分在远程计算机上执行、或者完全在远程计算机或服务器上执行。在涉及远程计算机的情形中,远程计算机可以通过任意种类的网络—包括局域网(LAN)或广域网(WAN)—连接到用户计算机,或者,可以连接到外部计算机(例如利用因特网服务提供商来通过因特网连接)。

[0211] 在此将参照根据本发明实施例的方法、装置(系统)和计算机程序产品的流程图和/或框图描述本发明的各个方面。应当理解,流程图和/或框图的每个方框以及流程图和/或框图中各方框的组合,都可以由计算机程序指令实现。这些计算机程序指令可以提供给通用计算机、专用计算机或其它可编程数据处理装置的处理器,从而生产出一种机器,使得这些指令在通过计算机或其它可编程数据处理装置的处理器执行时,产生了实现流程图和/或框图中的一个或多个方框中规定的功能/动作的装置。

[0212] 也可以把这些计算机程序指令存储在计算机可读介质中,这些指令使得计算机、其它可编程数据处理装置、或其它设备以特定方式工作,从而,存储在计算机可读介质中的指令就产生出包括实现流程图和/或框图中的一个或多个方框中规定的功能/动作的指令的制造品(article of manufacture)。

[0213] 也可以把计算机程序指令加载到计算机、其它可编程数据处理装置、或其它设备上,使得在计算机、其它可编程装置或其它设备上执行一系列操作步骤,以产生计算机实现的过程,从而使得在计算机或其它可编程装置上执行的指令提供实现流程图和/或框图中的一个或多个方框中规定的功能/动作的过程。

[0214] 附图中的流程图和框图显示了根据本发明的不同实施例的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上,流程图或框图中的每个方框可以代表一个模块、程序段或代码的一部分,所述模块、程序段或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应当注意,在有些作为替换的实现中,方框中所标注的功能可以以不同于附图中所标注的顺序发生。例如,两个连续的方框实际上可以基本

并行地执行,它们有时也可以按相反的顺序执行,这依所涉及的功能而定。也要注意的,框图和/或流程图中的每个方框、以及框图和/或流程图中的方框的组合,可以用执行规定的功能或动作的专用的基于硬件的系统来实现,或者可以用专用硬件与计算机指令的组合来实现。

[0215] 应当注意,在此描述的任何方法都可以包括提供一种系统的额外步骤,所述系统包括包含在计算机可读存储介质中的不同软件模块;所述模块例如可以包括在框图中示出和/或在此描述的任何或全部元素;作为实例而非限制,图5和19中所示的任何一个、部分或全部模块、子模块和/或数据库。然后可以使用所述系统的不同软件模块和/或子模块(如上所述,在一个或多个硬件处理器16上执行)执行所述方法步骤。进一步,计算机程序产品可以包括计算机可读存储介质,其具有适合于被执行的代码以便执行在此描述的一个或多个方法步骤,包括为所述系统提供不同软件模块。

[0216] 在任何情况下,应当理解,在此示出的组件可以以各种形式的硬件、软件或它们的组合来实现;例如,专用集成电路(多个)(ASIC)、功能电路、一个或多个具有关联存储器的经过适当编程的通用数字计算机等。给予了在此提供的本发明的教导后,本领域的技术人员将能够构想本发明的组件的其它实现。

[0217] 在此使用的术语只是为了描述特定的实施例并且并非旨在作为本发明的限制。如在此使用的,单数形式“一”、“一个”和“该”旨在同样包括复数形式,除非上下文明确地另有所指。还将理解,当在此说明书中使用时,术语“包括”和/或“包含”指定了声明的特性、整数、步骤、操作、元素和/或组件的存在,但是并不排除一个或多个其它特性、整数、步骤、操作、元素、组件和/或其组合的存在或增加。

[0218] 下面权利要求中的对应结构、材料、操作以及所有功能性限定的装置或步骤的等同替换,旨在包括任何用于与在权利要求中具体指出的其它元件相组合地执行该功能的结构、材料或操作。出于示例和说明目的给出了对本发明的描述,但所述描述并非旨在是穷举的或是将本发明限于所公开的形式。在不偏离本发明的范围和精神的情况下,对于所属技术领域的普通技术人员来说许多修改和变化都将是显而易见的。实施例的选择和描述是为了最佳地解释本发明的原理和实际应用,并且当适合于所构想的特定使用时,使得所属技术领域的其它普通技术人员能够理解本发明的具有各种修改的各种实施例。

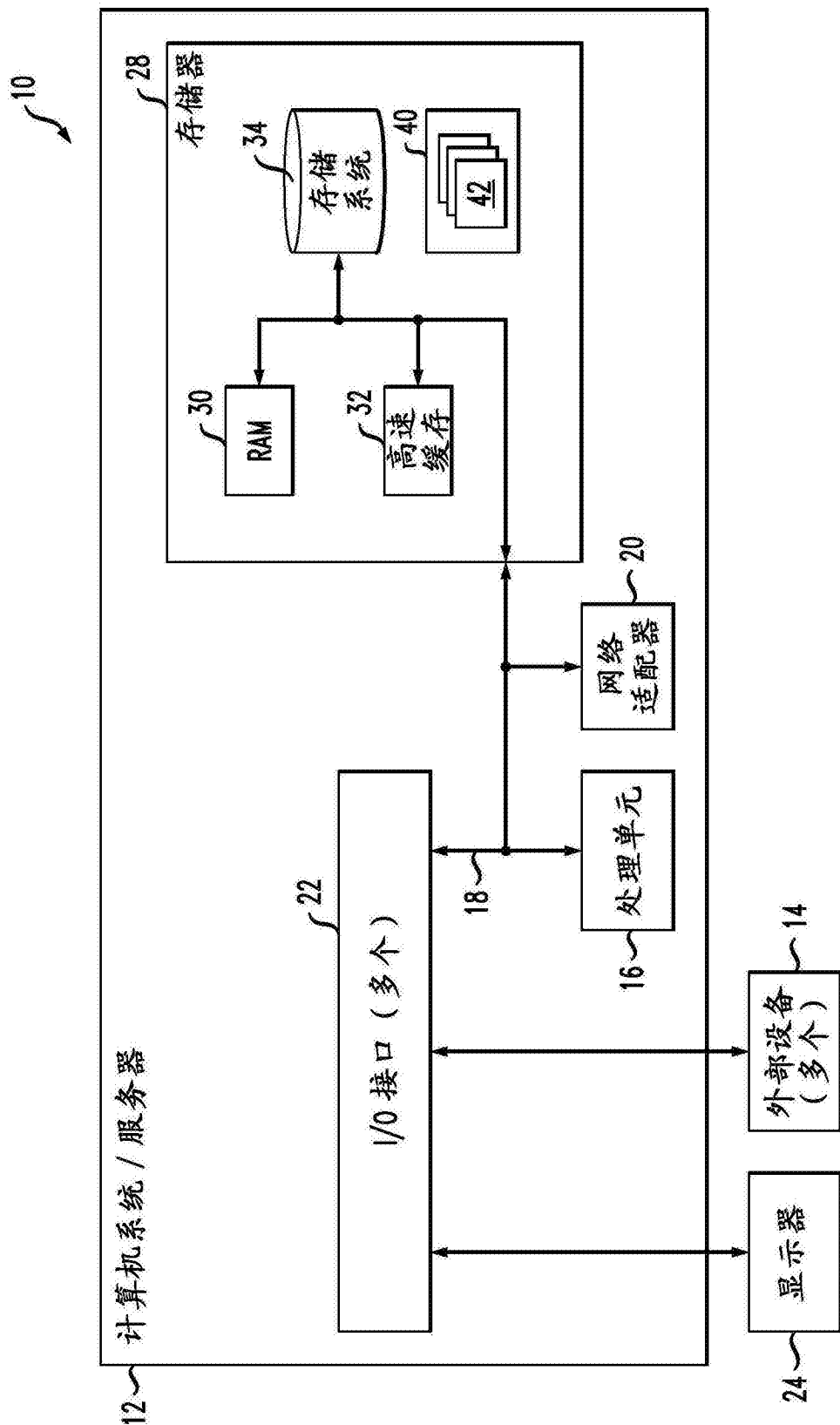


图1



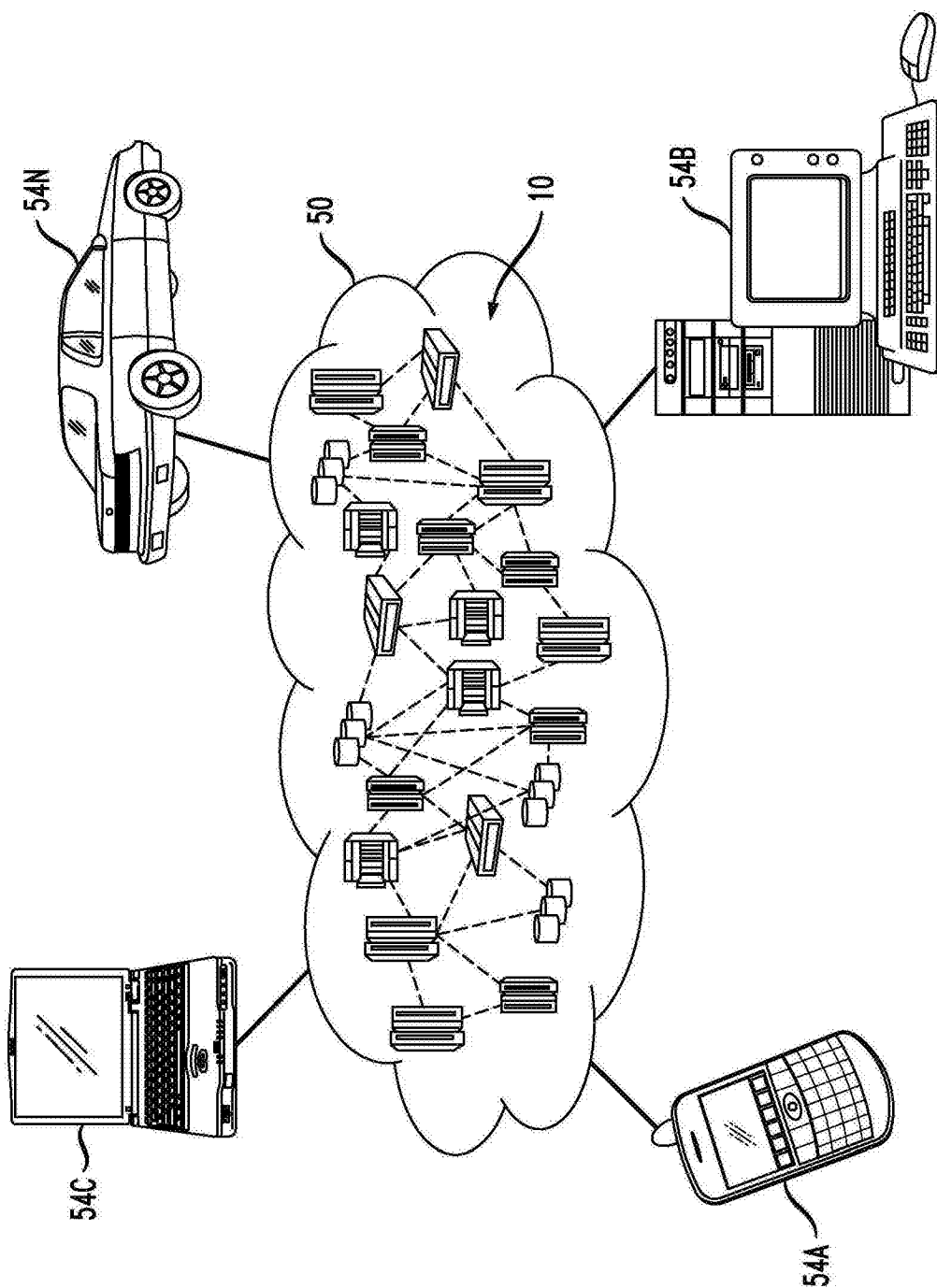


图2

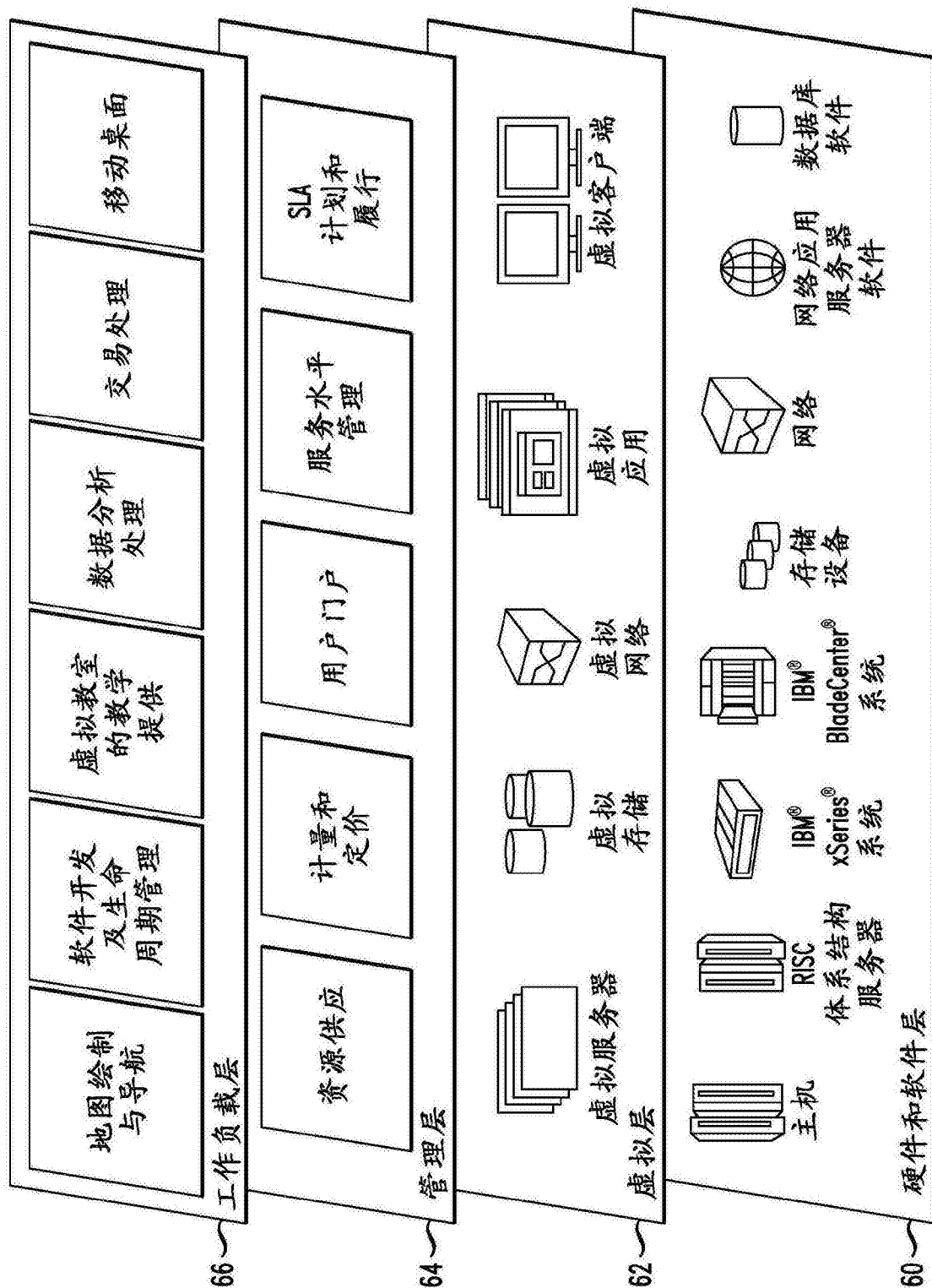


图3

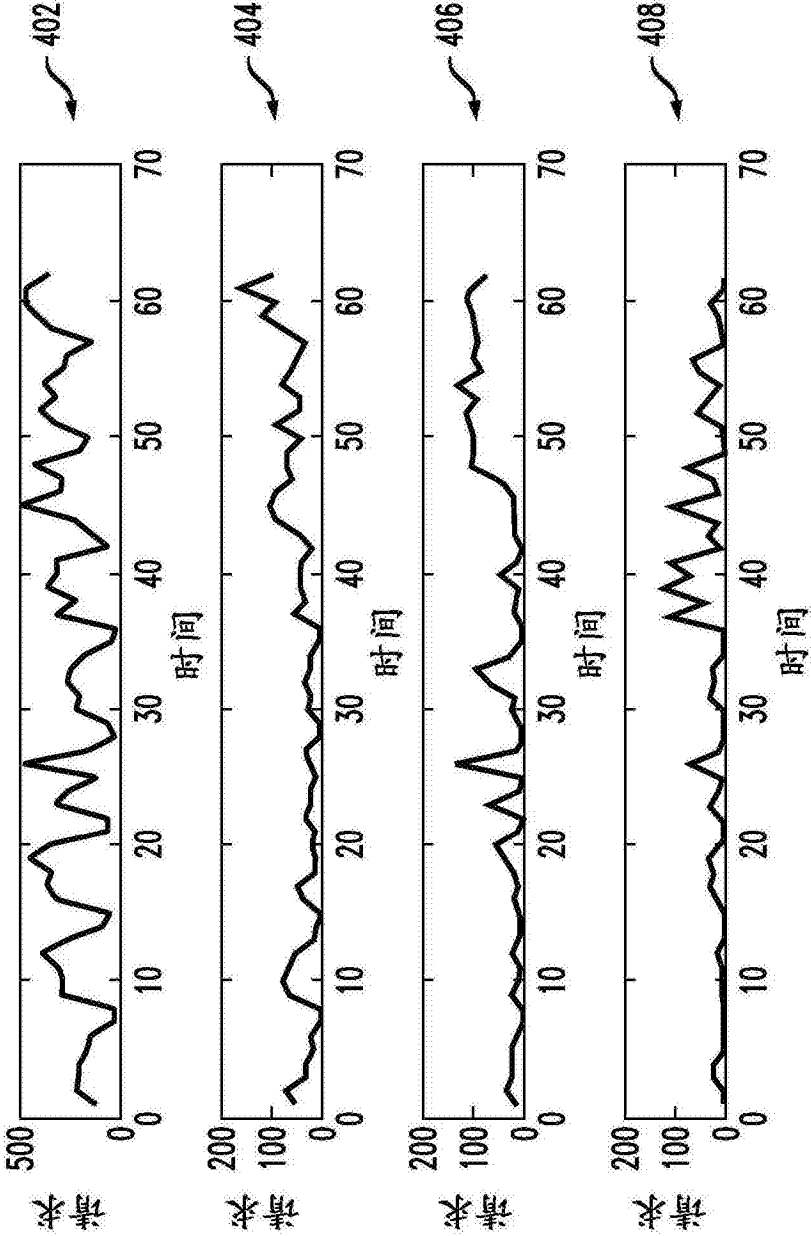


图4

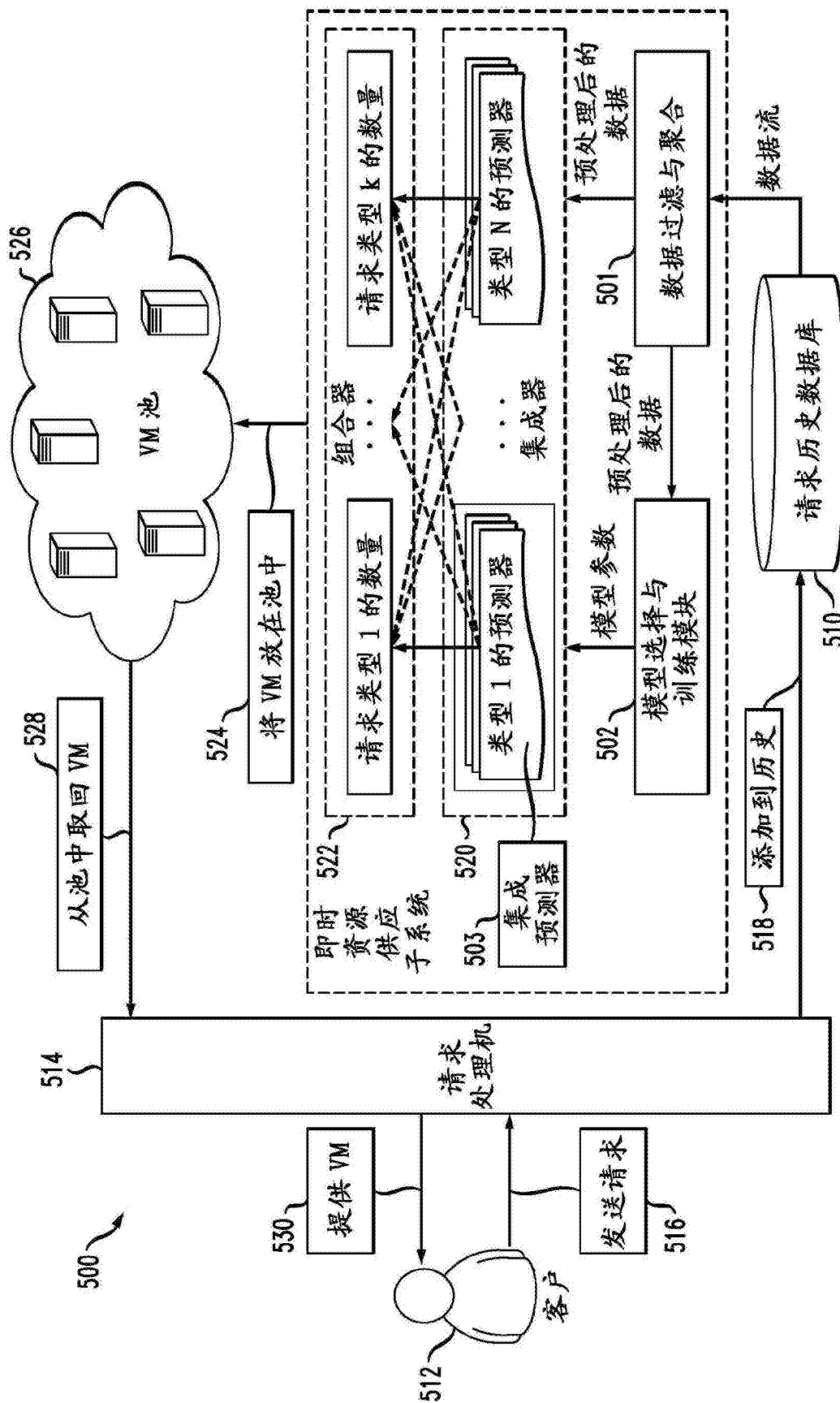


图5

云预测成本的  
符号

符号	定义
$v(t)/\hat{v}(t)$	VM 在时间 t 处的实际 / 预测的需求
$T(v(t), \hat{v}(t))$	服务质量的成本函数
$R(v(t), \hat{v}(t))$	闲置资源的成本函数
$T_{hit}/T_{miss}$	准备 / 未准备的 VM 的供应时间
$R_{vm}$	VM 在单位时间内的资源成本
$\beta$	服务质量成本在总成本中的权重

图6

时间序列预测  
算法

预测方法	描述
移动平均	单纯
自回归	线性回归
人工神经网络	非线性回归
支持向量机	具有非线性内核的线性学习器
基因表达式编程	试探算法

图7

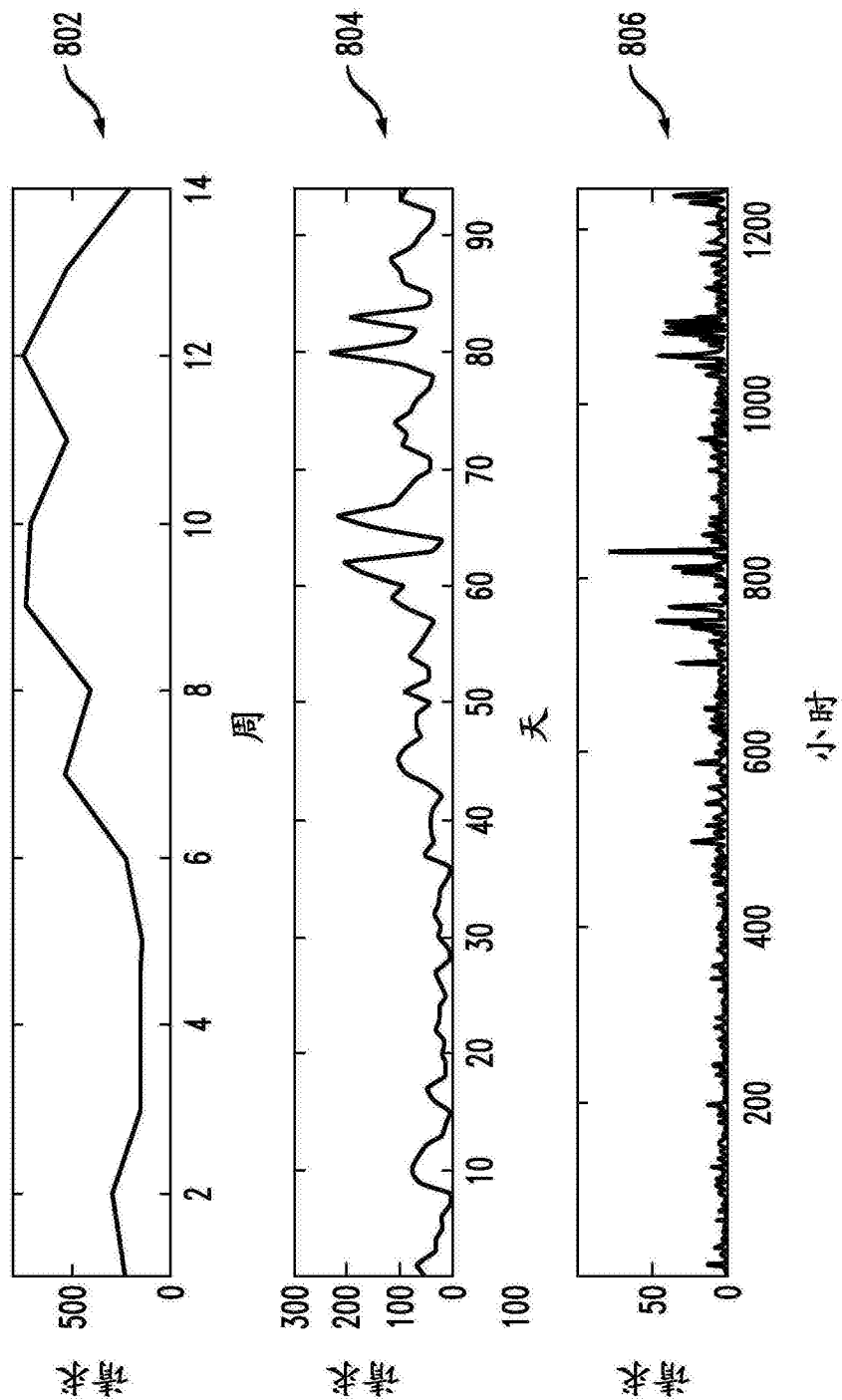


图8

VM 需求时间序列的特征

测量	每周	每天	每小时
变异系数	0.5606	0.7915	1.2249
偏度	0.3295	1.5644	5.4464
峰度	1.62	5.8848	52.4103

图9

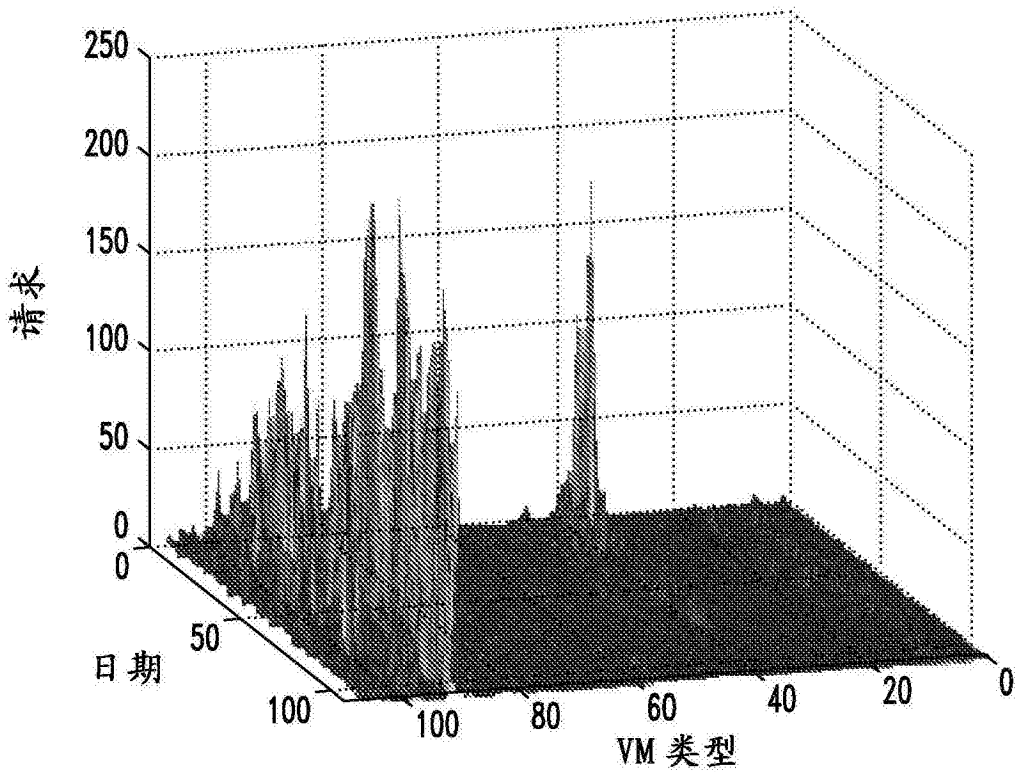


图10



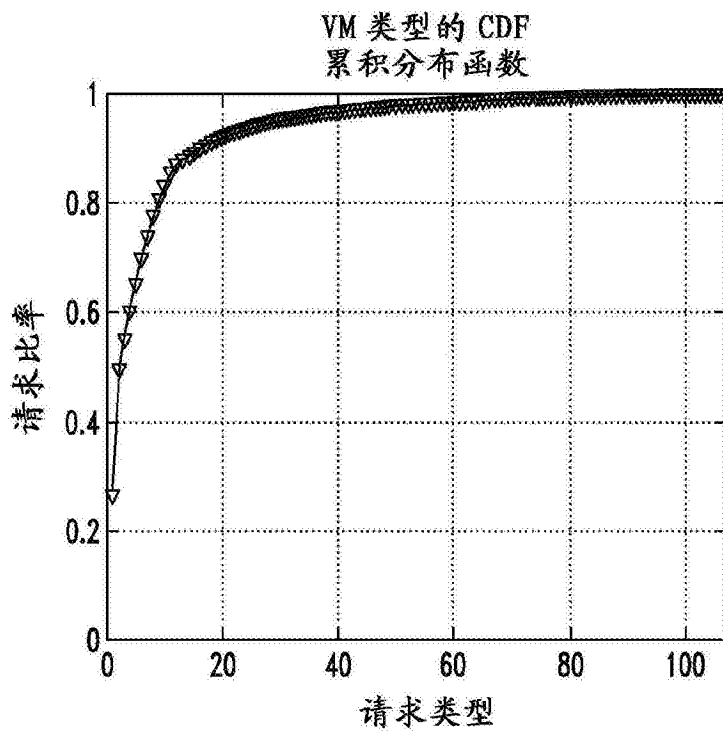


图11A

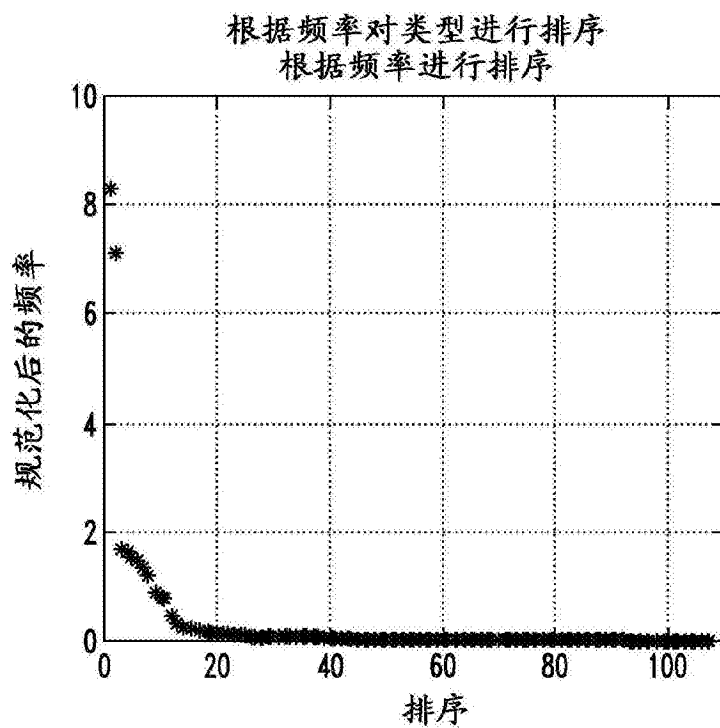


图11B

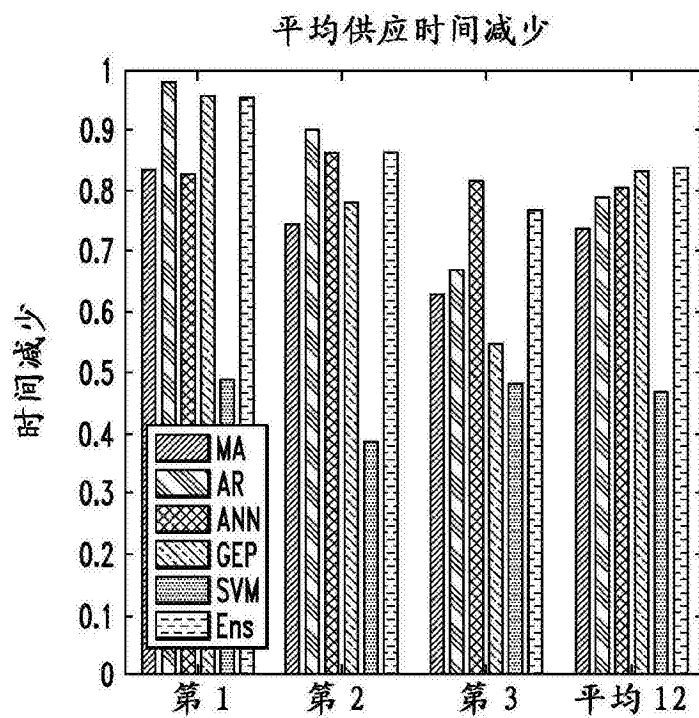


图12A

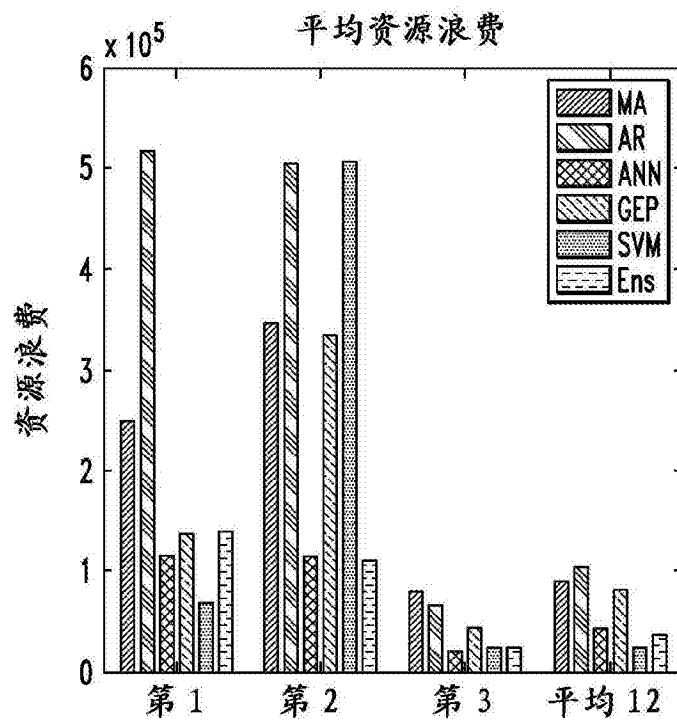


图12B

不同测量下的预测算法成本														
预测器	1302 ↙				1304 ↙				1306 ↙				1308 ↙	
	第一 VM 类型				第二 VM 类型				第三 VM 类型				前 12 的平均	
	MAE	MSE	MAPE	CPC	MAE	MSE	MAPE	CPC	MAE	MSE	MAPE	CPC		
MA	40.3	2740.23	0.87	437305	51.53	4904.4	1.63	532132.5	7.23	88.5	1.67	76370	152402.29	
AR	45.1	4034.57	1.52	300687.5	50.5	3893.63	5.32	391630	5.9	67.5	1.05	83292.5	137046.67	
ANN	30.27	1203.07	0.67	388535	20.8	706.4	2.04	247085	4.97	29.57	2.14	37052.5	106113.96	
GEP	17.37	1757.5	0.21	154440	54.77	5621.97	6.59	473347.5	4.1	27.63	1.59	33745	112149.79	
SVM	68.03	5604.77	0.9	1010422.5	59.5	5578.37	2.89	890447.5	6.2	85.4	0.9	88737.5	234225.83	
集成	16.7	1212.1	0.21	158862.5	21.67	1057	2.04	254190	5.03	46.97	1.56	53327.5	88679.79	

图13

## RED HAT ENTERPRISE LINUX 5.5 (32 位)

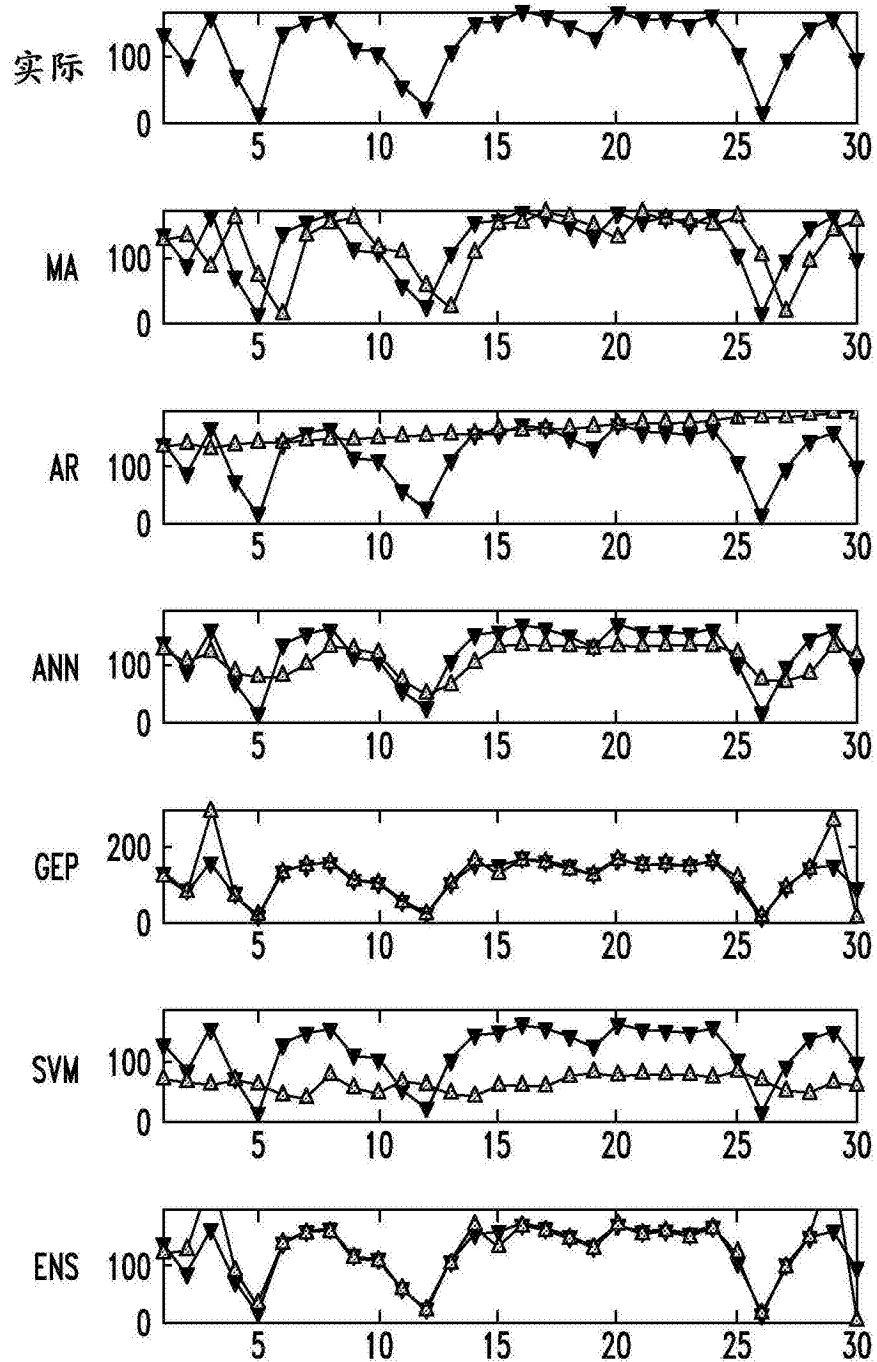


图14A

## RED HAT ENTERPRISE LINUX 5.5 (64位)

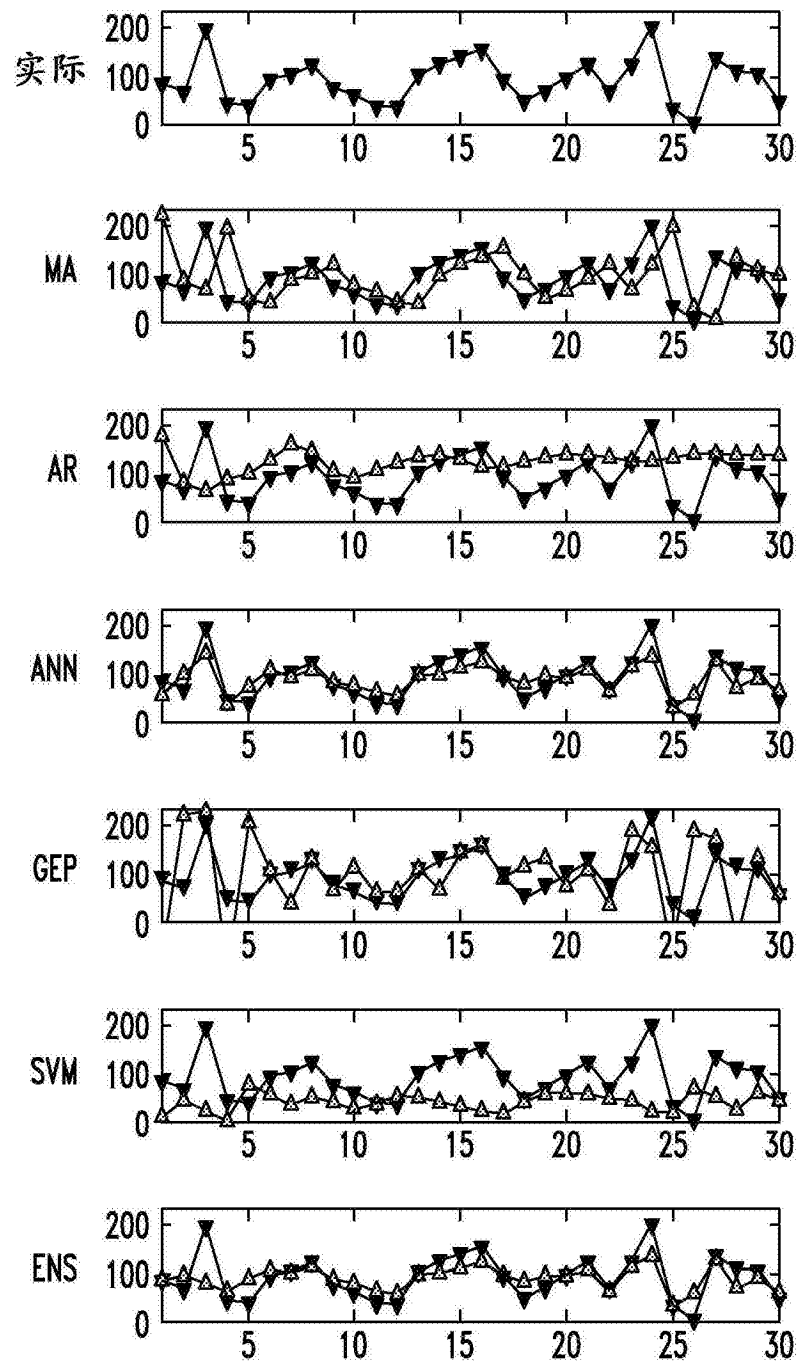


图14B

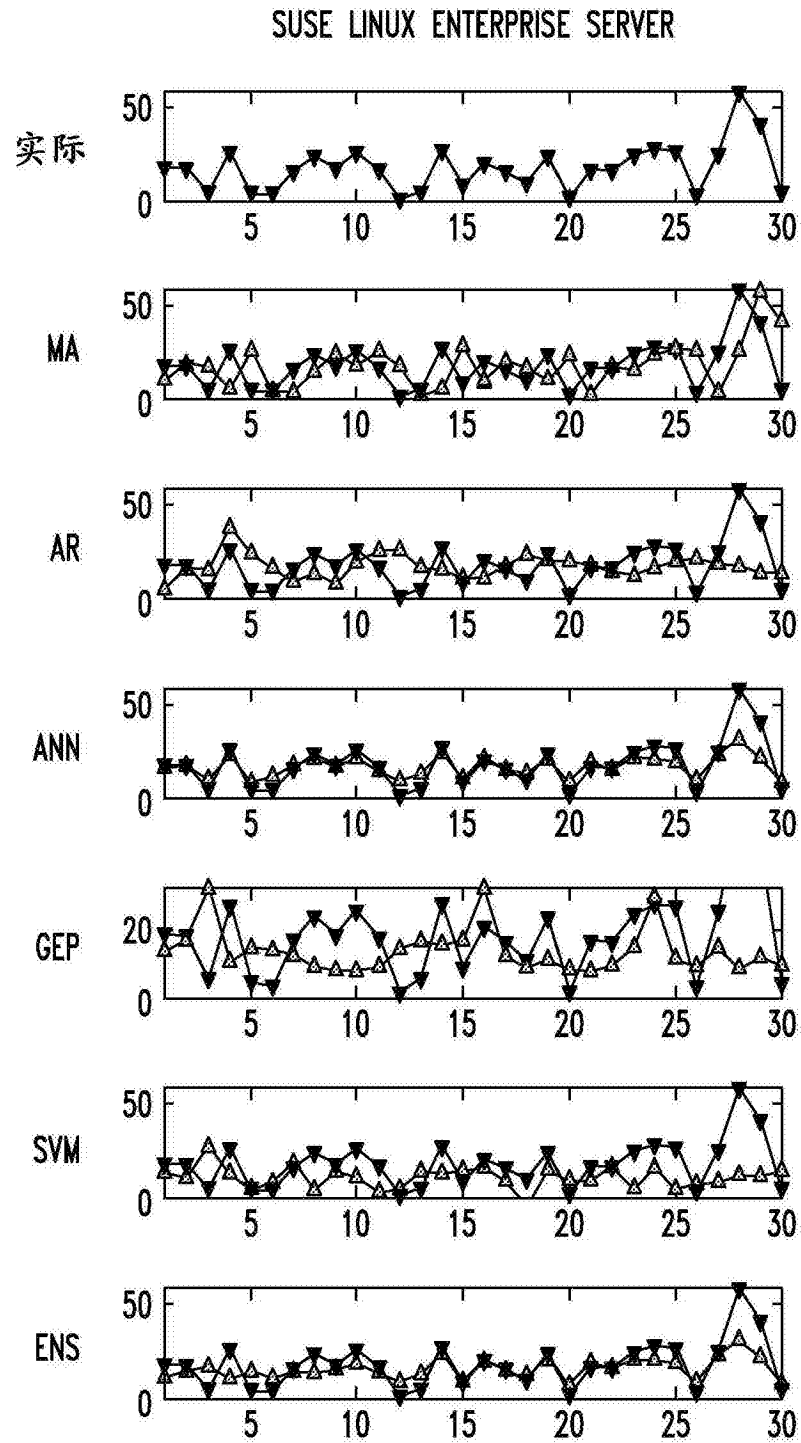


图14C

不同成本测量指标的预测器的等待时间减少和资源浪费

时间减少	平均时间减少	平均资源浪费
MAE	81.77%	28300
MSE	80.13%	37100
MAPE	79.01%	33133
CPC	83.79%	38533

图15

通过结合预留控制器实现的供应时间减少

VM 类型	1	2	3	4	5	6
减少	97.77%	91.58%	96.81%	96.26%	96.76%	91.67%
VM 类型	7	8	9	10	11	12
减少	98.17%	94.37%	99.56%	85.35%	84.98%	97.45%

图16



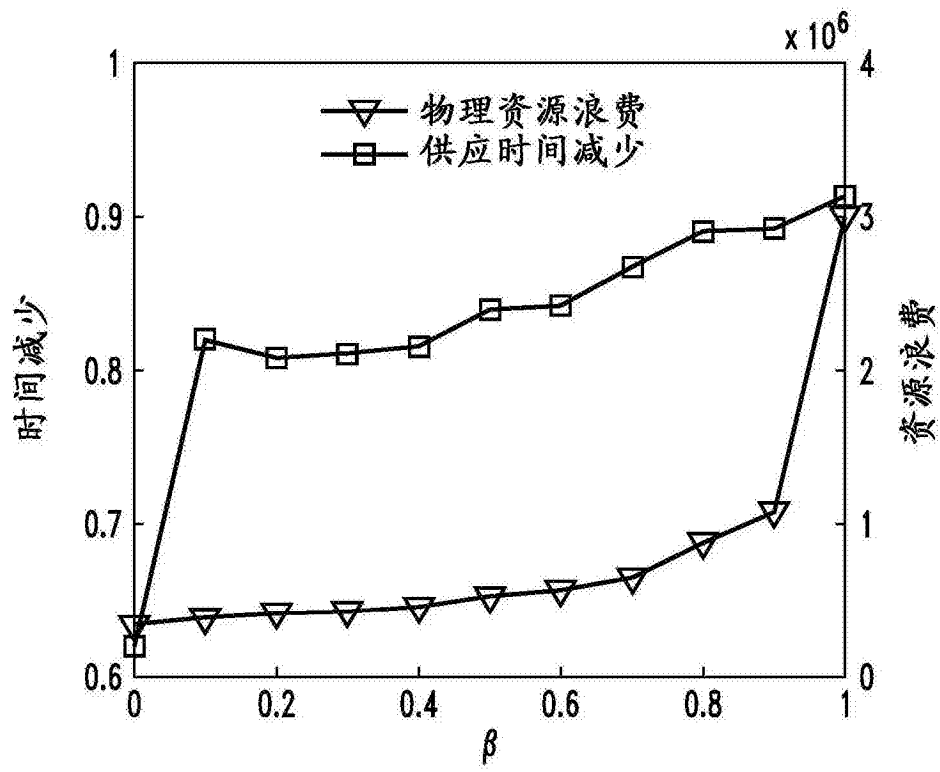
$\beta$  效应

图17

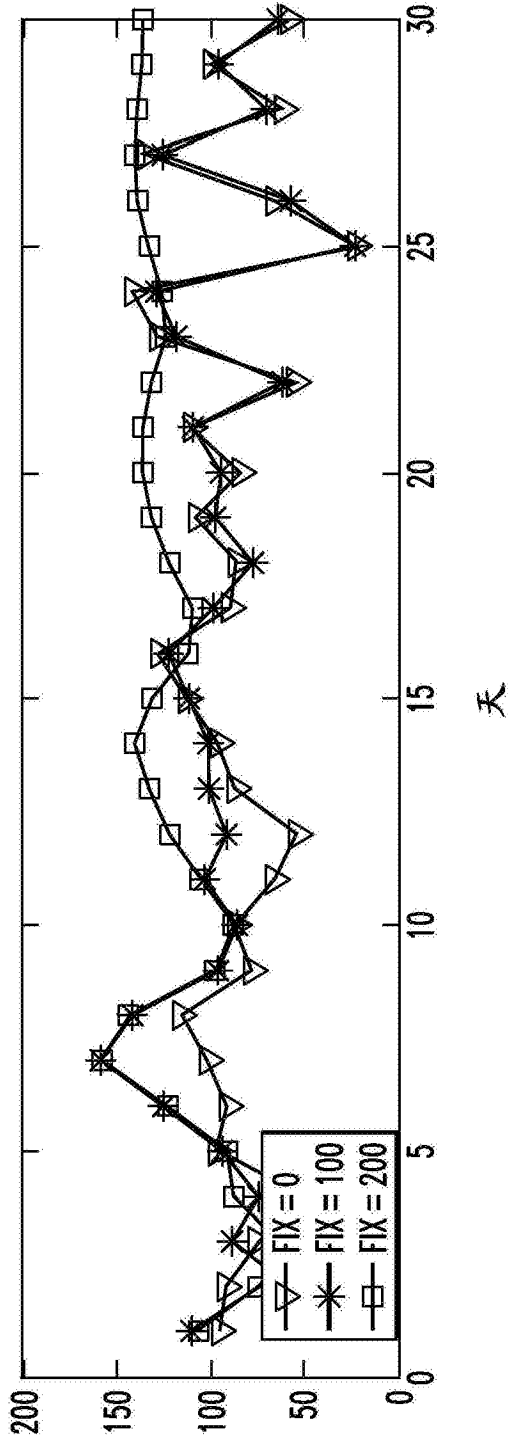


图18

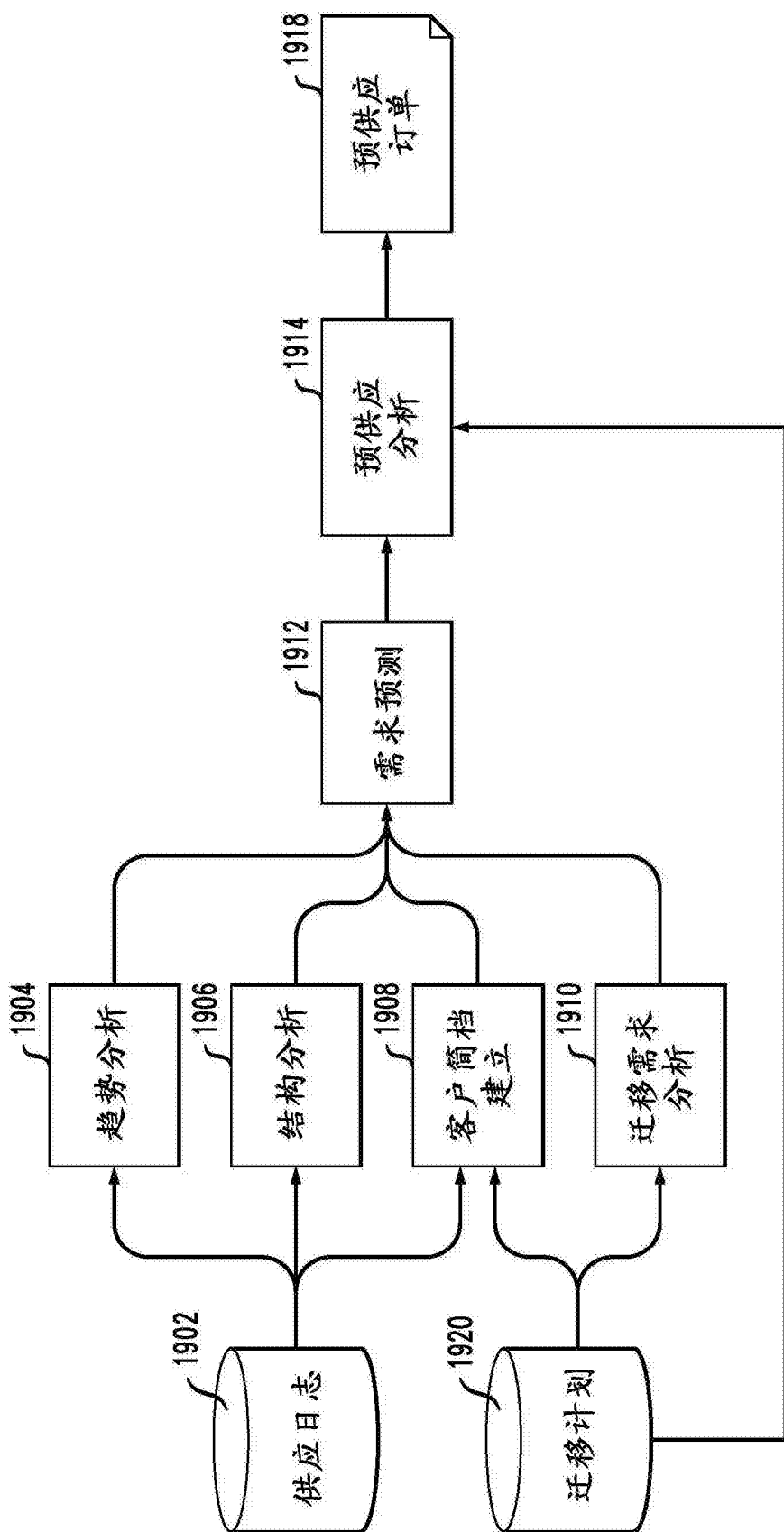


图19